

UNIVERSIDAD POLITÉCNICA SALESIANA
SEDE QUITO

CARRERA:
INGENIERÍA ELECTRÓNICA

Trabajo de titulación previo a la obtención del título de:
INGENIERO ELECTRÓNICO

TEMA:
DESARROLLO DE UN ROBOT DELTA PARA APLICACIONES DE
MANIPULACIÓN DE OBJETOS Y VISIÓN ARTIFICIAL MEDIANTE EL
SENSOR KINECT

AUTOR:
JORGE ESTEBAN MARTÍNEZ MACANCELA

TUTOR:
VÍCTOR VINICIO TAPIA CALVOPÍÑA

Quito, noviembre del 2018

CESIÓN DE DERECHOS DE AUTOR

Yo Jorge Esteban Martínez Macancela, con documento de identificación N° 1723622153, manifiesto mi voluntad y cedo a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que soy autor del trabajo de grado/titulación intitulado: “DESARROLLO DE UN ROBOT DELTA PARA APLICACIONES DE MANIPULACIÓN DE OBJETOS Y VISIÓN ARTIFICIAL MEDIANTE EL SENSOR KINECT”, mismo que ha sido desarrollado para optar por el título de: Ingeniero Electrónico, en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En aplicación a lo determinado en la Ley de Propiedad Intelectual, en mi condición de autor me reservo los derechos morales de la obra antes citada. En concordancia, suscribo este documento en el momento que hago entrega del trabajo final en formato impreso y digital a la Biblioteca de la Universidad Politécnica Salesiana.



.....
Nombre: Jorge Esteban Martínez Macancela

Cédula: 1723622153

Fecha: noviembre del 2018

DECLARATORIA DE COAUTORÍA DEL DOCENTE TUTOR

Yo declaro que bajo mi dirección y asesoría fue desarrollado el trabajo de titulación:
“DESARROLLO DE UN ROBOT DELTA PARA APLICACIONES DE
MANIPULACIÓN DE OBJETOS Y VISIÓN ARTIFICIAL MEDIANTE EL
SENSOR KINECT” realizado por Jorge Esteban Martínez Macancela, obteniendo un
producto que cumple con todos los requisitos estipulados por la Universidad
Politécnica Salesiana para ser considerados como trabajo final de titulación.

Quito, noviembre del 2018



Víctor Vinicio Tapia Calvopiña

C.I. 1708547219

DEDICATORIA

La fuerza, la perseverancia y el conocimiento brindados por Dios me ha traído a este punto, todo es por él y para él y es por eso que le dedico este trabajo con todo mi ser, a mis padres que siempre sacrificaron su felicidad para que yo pueda superarme convirtiéndose en una inspiración latente día a día para cumplir mis metas y nunca rendirme, sobre todo a mi madre Bertha una mujer con un amor inmenso hacia mí y mis hermanos que siempre peleo por nosotros para que podamos ser felices en todo momento, a los miembros de mi familia, mi hermano Fernando mi compañero de juegos que nunca se olvide que todo se puede conseguir aunque se tarde para lograrlo siempre estaré para ti, a mi hermana Iveth que siempre dio su cariño, consejos y ejemplo como una extraordinaria hermana mayor, madre de familia y amiga que nunca nos abandona, mi cuñado Esteban que se ganó el aprecio y voluntad de mí y mi familia y siempre está pendiente de mi todos nosotros y por último pero no menos importante a mi adorada sobrina Anna Florencia que llena de cariño nuestro hogar y corazones nunca te rindas, te quiero mucho pequeña.

También dedico este trabajo a todos mis amigos que siempre están apoyándome, aconsejándome y velando por mi seguridad, Andrés A, Andrés R, Adriana, Eduardo, Fabricio, Bryan, Johanna, Vladimir, David, Luis, Andrea, Verónica y Paúl gracias por hacer esta vida más llevadera, sobre todo a ti amigo Paul ya que eres un hermano para mí, gracias por tus ocurrencias, siempre los llevo conmigo.

Nunca se debe perder la fe en que se puede ganar, el autoconvencimiento le da a uno el poder de cambiar el destino.

Jorge Esteban Martínez Macancela

AGRADECIMIENTO

Sobre todo, quiero dar gracias a Dios por darme la entereza ante las adversidades que enfrente, por permitirme culminar una etapa más de la vida con inmensas satisfacciones y bendiciones, sobre todo por tu amor, gracias por permitirme seguir vivo, porque todas las cosas proceden de él y para él, y existen por él y para él.

A mi madre que no me bastaran las palabras y calificativos para alagarla siempre a mi lado brindándome todo su cariño para ser el hombre que soy ahora, corrigiendo mis malos hábitos y por qué siempre cada vez que me mira me llena de fuerza y coraje para seguir adelante, gracias viejita bella.

Agradezco al Ing. David Espín que supo orientar el ámbito profesional y estudiantil de mi persona de una manera correcta, gracias por confiar en mi amigo, al Ing. Christian Crisanto que con su influencia me permitió confiar en mis virtudes y al MSc. Vladimir Araujo que me otorgó los conocimientos para la elaboración de este proyecto y siempre me presiona para convertirme en alguien mejor, gracias amigo.

A los Ingenieros: Junior Figueroa, Alejandra Fernández y Vinicio Tapia que fueron mis tres tutores que me han acompañado en diferentes etapas en el transcurso de la realización de este proyecto los cuales crearon senderos para poder seguirlos, proporcionaron consejos de forma oportuna y me otorgaron de su tiempo de forma desinteresada para poder culminar este proyecto de manera exitosa

ÍNDICE DE CONTENIDO

CESIÓN DE DERECHOS DE AUTOR.....	i
DECLARATORIA DE COAUTORÍA DEL DOCENTE TUTOR.....	ii
DEDICATORIA	iii
AGRADECIMIENTO	iv
ÍNDICE DE CONTENIDO	v
ÍNDICE DE FIGURAS.....	ix
ÍNDICE DE TABLAS	xii
RESUMEN	xiii
ABSTRACT.....	xiv
INTRODUCCIÓN	xv
CAPÍTULO 1	1
ANTECEDENTES	1
1.1 Tema.....	1
1.2 Planteamiento del Problema.....	1
1.3 Justificación del Trabajo	1
1.4 Objetivos	2
1.4.1 Objetivo general.....	2
1.4.2 Objetivos específicos	2
CAPÍTULO 2	3
FUNDAMENTACIÓN TEÓRICA	3
2.1 Robots Paralelos	3
2.1.1 Robot Delta.....	4
2.1.2 Actuadores finales	5
2.1.2.1 Ventosa de Vacío.....	6
2.1.2.2 Pinza de Presión o Gripper	6

2.1.2.3 <i>Electroimán</i>	7
2.2 Cinemática de un Robot Paralelo	7
2.2.1 Modelo Cinemático General de un Robot Delta.....	7
2.2.2 Cinemática Inversa	12
2.3 Cinemática general de una Pinza o Gripper	15
2.4 Sensor Kinect	18
2.5 Processing.....	20
2.6 Polímeros de acrilonitrilo-butadieno-estireno (plástico ABS)	21
CAPÍTULO 3	23
DESARROLLO Y CONSTRUCCIÓN DEL HARDWARE Y DEL SOFTWARE .	23
3.1 Visión General.....	23
3.2 Hardware Mecánico	23
3.2.1 Diseño de Piezas en Autocad 2015.....	24
3.2.2 Base Fija	24
3.2.2.1 <i>Diseño de la base fija</i>	24
3.2.2.2 <i>Construcción de la base fija</i>	25
3.2.3 Aletas	26
3.2.3.1 <i>Diseño de las Aletas</i>	26
3.2.3.2 <i>Construcción de Aletas</i>	27
3.2.4 Brazos	28
3.2.4.1 <i>Diseño de los Brazos</i>	28
3.2.4.2 <i>Construcción de los Brazos</i>	29
3.2.5 Base Móvil.....	30
3.2.5.1 <i>Diseño de la base Móvil</i>	30
3.2.5.2 <i>Construcción de la Base Móvil</i>	30
3.2.6 Gripper o Pinza.....	31
3.2.6.1 <i>Diseño de la pinza o Gripper</i>	31

3.2.6.2 Construcción de la pinza o Gripper.....	32
3.3 Ensamblaje Total.....	32
3.4 Desarrollo del Hardware Electrónico.....	33
3.4.1 Esquema General del Sistema	34
3.4.2 Tarjeta de Actuación Maestro Arduino.	34
3.4.3 Tarjeta de Control Auxiliar de alimentación	36
3.4.3.1 Diseño de la tarjeta de Control Auxiliar de alimentación.....	36
3.4.3.2 Construcción de la tarjeta de control auxiliar de alimentación.....	39
3.5 Desarrollo del Software.....	39
3.5.1 Diseño del Software.....	40
3.5.1.1 Tareas del Entorno de Processing	40
3.5.1.2 Tareas de la tarjeta de Control Arduino.....	40
3.5.1.3 Tareas de la tarjeta de control de alimentación Auxiliar.....	40
3.5.2 Implementación del Software	40
3.5.2.1 Diagrama general de flujo del algoritmo del Robot Delta.....	41
3.5.2.2 Análisis Cinemático Inverso a nivel de Software	42
3.5.2.3 Diagrama de flujo para el control del Robot.....	44
3.5.2.4 Diagrama de flujo para el control de los brazos o Piernas	45
3.5.2.5 Diagrama de flujo para Kinect y Delta	45
3.5.2.6 Apertura y cerrado de Gripper o pinza	46
3.5.2.7 Diagrama de flujo rutina de Gripper.....	48
3.5.2.8 Tarjeta de Control Arduino	48
CAPÍTULO 4.....	50
ANÁLISIS DE RESULTADOS	50
4.1 Pruebas de Hardware Mecánico	50
4.2 Pruebas de Tarjeta de Control	51
4.2.1 Comunicación Sensor Kinect-Processing.....	51

4.2.2 Comunicación Processing – Tarjeta de Actuación Arduino.....	53
4.3 Protocolos de Pruebas de Movimiento.....	54
4.3.1 Movimiento Vertical.....	55
4.3.2 Movimiento Horizontal	58
4.3.3 Movimiento de Profundidad	60
4.4 Pruebas de precisión de empacado de objetos.	62
4.4.1 Prueba de precisión de empacado en un punto X1 del área del trabajo	63
4.4.2 Prueba de precisión de empacado en un punto X2 del área del trabajo	64
4.4.3 Prueba de precisión de empacado en un punto X3 del área del trabajo	66
CONCLUSIONES	68
RECOMENDACIONES.....	70
REFERENCIAS BIBLIOGRÁFICAS.....	71
ANEXOS	73

ÌNDICE DE FIGÜRAS

Figura 2.1 Primer Robot Paralelo	3
Figura 2.2 Robot Hexápodo	4
Figura 2.3 Robot Delta.....	5
Figura 2.4 Ventosa de Vacío.....	6
Figura 2.5 Pinza de Presión O Gripper	6
Figura 2.6 Electroimán.....	7
Figura 2.7 Estructura del Robot Paralelo	8
Figura 2.8 Robot Delta y su modelo geométrico	9
Figura 2.9 Descripción de los ángulos: izq) Vista frontal, der) Vista lateral.....	9
Figura 2.10 forma geométrica de: a) la base fija, b) plataforma móvil.....	12
Figura 2.11 Representación vectorial de la primera cadena cinemática	12
Figura 2.12 Tipos de carga puntual debido a la sujeción.....	15
Figura 2.13 Función simple de Sujeción.....	16
Figura 2.14 Representación gráfica de las fuerzas de contacto para mandíbulas.	17
Figura 2.15 Secciones principales del Sensor Kinect	19
Figura 2.16 Composición interna del Sensor Kinect	19
Figura 2.17 Distancias de detección modo por defecto y modo near	19
Figura 2.18 Descripción de captura de movimientos del Sensor Kinect	20
Figura 2.19 Logo de inicio de software Processing 2.2.1	21
Figura 2.20 Plástico ABS presente en el mercado	22
Figura 3.1 Modelos de diferentes robots delta en el mercado	23
Figura 3.2 Diseño de la base Fija del Robot Delta	24
Figura 3.3 Base Fija del Robot Delta acoplamiento	25
Figura 3.4 Base Fija del Robot Delta acoplamiento	25
Figura 3.5 Base Fija del Robot Delta implementada	26

Figura 3.6 Diseño de aletas	27
Figura 3.7 Aletas implementadas mediante impresión 3D	27
Figura 3.8 Acople de ruliman en aleta Impresa en 3D.....	28
Figura 3.9 Diseño de Brazos en AutoCad2015.....	29
Figura 3.10 Diseño de Brazos en AutoCad2015.....	29
Figura 3.11 Diseño de Base móvil en AutoCad2015.....	30
Figura 3.12 Base móvil impresa	31
Figura 3.13 Pinza Diseñada en el entorno de AutoCAD	31
Figura 3.14 Pinza Ensamblada.....	32
Figura 3.15 Robot Delta Parcial Diseñado en el entorno de AutoCAD	32
Figura 3.16 Robot Final Diseñado en el entorno de AutoCAD	33
Figura 3.17 Robot Final Delta implementado.....	33
Figura 3.18 Esquema General de control del sistema del Robot Delta.....	34
Figura 3.19 Diagrama electrónico Tarjeta Arduino Uno	35
Figura 3.20 Esquema electrónico Tarjeta de control de alimentación.....	37
Figura 3.21 PCB de la tarjeta de Control de alimentación.....	38
Figura 3.22 Tarjeta de Control auxiliar de alimentación terminada	39
Figura 3.23 Estructura General del software para el entorno Processing	41
Figura 3.24 Análisis geométrico	43
Figura 3.25 Diagrama de flujo para el algoritmo de control del robot	44
Figura 3.26 Diagrama de flujo para el algoritmo de control de los brazos.....	45
Figura 3.27 Diagrama de flujo para el algoritmo de control de los brazos.....	46
Figura 3.28 Nube de puntos en Nite(cloudPoints).....	47
Figura 3.29 Diagrama de flujo representado a la rutina de Gripper	48
Figura 3.30 Diagrama de flujo representado a la rutina de actuación en la tarjeta Arduino	49

Figura 4.1 Plataforma Delta	50
Figura 4.2 Interfaz de presentación de datos Capturados por el sensor Kinect	52
Figura 4.3 Interfaz de presentación de datos Capturados por el sensor Kinect para la pinza o gripper	53
Figura 4.4 Datos enviados de manera serial hacia la tarjeta de Actuación	54
Figura 4.5 Ejecución de Rutina de movimientos para el Robot Delta	54
Figura 4.6 Movimientos capturados por el Sensor	55
Figura 4.7 Protocolo de pruebas de movimiento en el eje Y	56
Figura 4.8 Protocolo de pruebas de movimiento en el eje X	59
Figura 4.9 Protocolo de pruebas de movimiento en el eje Z	
Pruebas de movimiento de profundidad. Elaborado por: Esteban Martínez.....	61
Figura 4.10 Prueba de precisión 1 en un punto del área de trabajo	63
Figura 4.11 Prueba de precisión 2 en un punto del área de trabajo	64
Figura 4.12 Prueba de precisión 3 en un punto del área de trabajo	66

ÍNDICE DE TABLAS

Tabla 3.1 Pines de conexiones de la tarjeta Arduino Uno	36
Tabla 3.2 Pines de conexiones positivas de la tarjeta de control de Alimentación....	38
Tabla 3.3 Pines de conexiones negativas de la tarjeta de control de Alimentación...	39
Tabla 4.1 Valores obtenidos en la trayectoria trazada en el eje Y.	56
Tabla 4.2 Tabla de efectividad y errores en la trayectoria trazada en el eje Y.	57
Tabla 4.3 Valores obtenidos en la trayectoria trazada en el eje X.	59
Tabla 4.4 Tabla de efectividad y errores en la trayectoria trazada en el eje X.	60
Tabla 4.5 Valores obtenidos en la trayectoria trazada en el eje Z.	61
Tabla 4.6 Tabla de efectividad y errores en la trayectoria trazada en el eje Z.....	62
Tabla 4.7 Valores obtenidos en La prueba de precisión punto X1	63
Tabla 4.8 Tabla resumen de porcentajes de resultado de prueba 1	64
Tabla 4.9 Valores obtenidos en La prueba de precisión punto X2	65
Tabla 4.10 Tabla resumen de porcentajes de resultado de prueba 2.....	65
Tabla 4.11 Valores obtenidos en La prueba de precisión punto X3	66
Tabla 4.12 Tabla resumen de porcentajes de resultado de prueba 3.....	67
Tabla 4.13 Tabla resumen de porcentajes de resultado de las 3 pruebas de acierto en un solo punto.....	67

RESUMEN

En esta sección se detalla el desarrollo la construcción y las características técnicas de operación de los actuadores, tarjetas de control, sensores y software especial a utilizar en el robot tipo paralelo Delta. La plataforma Delta está formada por una plataforma fija, una plataforma móvil y una pinza que en conjunto poseen 5 grados de libertad la cual es completamente maniobrable si así se lo demanda. El prototipo está basado en el Robot comercial LLC army Delta, el robot tiene como especialidad el posicionamiento de objetos mediante la réplica de movimientos de la mano del usuario.

El robot Delta estará controlado mediante el sensor Kinect de Xbox versión 1.14, que operará mediante la imitación de movimientos capturados por el sensor usando visión artificial, aportando control y manejo en tiempo real esperando un fácil manejo del mismo. La implementación del robot Delta empieza por ingeniería inversa de su diseño tomando como referencia al LLC army Delta, para la construcción se realizó el método de impresión 3D diseñando cada pieza en el entorno AutoCad 2016.

Se realizarán las pertinentes pruebas de comunicación, y operación para comprobar la precisión del robot, una vez alcanzado los objetivos el robot procura servir de inspiración en aplicaciones de diferentes métodos de control en diferentes aplicaciones robóticas.

ABSTRACT

This section details the development of the construction and the technical characteristics of operation of the actuators, control cards, sensors and special software to be used in the Delta parallel type robot. The Delta platform consists of a fixed platform, a mobile platform and a clamp that together have 5 degrees of freedom which is completely maneuverable if requested. The prototype is based on the commercial robot LLC army Delta, the robot has as a specialty the positioning of objects through the replication of movements of the user's hand.

The Delta robot will be controlled by the Xbox Kinect sensor version 1.14, which will operate by imitating movements captured by the sensor using artificial vision, providing control and management in real time waiting for easy handling. The implementation of the Delta robot starts by reverse engineering its design taking as reference the LLC army Delta, for the construction the 3D printing method was designed, designing each piece in the AutoCad 2016 environment.

The relevant communication tests will be carried out, and operation to verify the accuracy of the robot, once the objectives are reached, the robot tries to serve as inspiration in applications of different control methods in different robotic applications.

INTRODUCCIÓN

La robótica en la actualidad se ha venido desarrollando de una manera importante en los últimos años y es una de las ramas de ingeniería moderna más utilizada e investigada en la industria gracias a la evolución vertiginosa de diferentes ideas que pueden ser aplicadas al modelamiento desarrollo y construcción de prototipos de robots cada vez más sofisticados, precisos, y algunos que cuentan con la característica de inteligencia artificial. Para cumplir con todo esto no solo se requiere una sinergia de los componentes mecánicos y electrónicos utilizados en el ensamblaje del robot, también es necesario detallar los modelos matemáticos que permitan una cohesión entre el hardware y el software y permitan lograr un movimiento armónico o toma de decisiones propias.

En el presente se tienen robots divididos en diferentes clasificaciones algunas acordes a los términos industriales en los que son aplicados, otros en base de su uso especial y por último por sus sistemas los cuales son: seriales y paralelos, la presente tesis se enfoca en los robots paralelos, específicamente hablando el Robot paralelo tipo Delta

El Robot Delta proyecta agregar una mayor comprensión en el análisis y desarrollo de la robótica manipuladora, donde los estudiantes, profesores e investigadores generen nuevas propuestas o ideas, ya que consta con una maniobrabilidad y manipulación amigable que permita la labor del mismo.

CAPÍTULO 1

ANTECEDENTES

1.1 Tema

Desarrollo de un Robot Delta para aplicaciones de manipulación de objetos y visión artificial mediante el sensor Kinect.

1.2 Planteamiento del Problema

En el Ecuador este tipo de robots no son muy comercializados por ende su estudio tampoco es abordado, es decir, no son presentados como ejemplos de análisis en las materias que están orientadas a la Robótica, lo que impide la adquisición de conocimiento en cuanto a sus características, tales como, posición, traslado y alta rigidez. Por tal motivo se ve la necesidad del desarrollo de este tipo de herramientas para la automatización industrial con el fin de aumentar y fomentar los conocimientos en la rama de la Robótica dentro del entorno educativo y para posteriormente ser aplicado en el campo industrial.

1.3 Justificación del Trabajo

Este proyecto propone la construcción de una herramienta de automatización que permita la manipulación de objetos, por medio de la captura y réplica del movimiento de las manos por parte de un robot paralelo y así desempeñar tareas de posicionamiento de objetos.

La implementación de este robot ayudará a interpretar correctamente su cinemática, para luego desarrollar un estudio de su comportamiento, mostrando cómo la aplicación de este resultaría productiva en la automatización industrial, a su vez podrá ser empleado en aplicaciones específicas y de esta forma al momento de desarrollar investigaciones futuras se tenga un indicio sobre cuál camino seguir y qué métodos se utilizarían para aumentar la eficiencia y robustez del robot.

1.4 Objetivos

1.4.1 Objetivo general

- Desarrollar un robot delta controlado mediante una aplicación con visión artificial y el sensor Xbox Kinect, para realizar la manipulación y posicionamiento de objetos en un lugar determinado.

1.4.2 Objetivos específicos

- Definir el tipo de cinemática a utilizar en la movilización y maniobrabilidad del robot Delta para luego ser estudiada y posteriormente simulada con lo cual se verificará que los movimientos estén acordes a los requisitos de funcionamiento especificados.
- Construir el robot delta para realizar posteriormente aplicaciones de manipulación de objetos.
- Desarrollar una aplicación con el software Processing que permita capturar los movimientos de las manos de un usuario mediante el sensor Kinect y que sean replicados sobre el robot delta utilizando para ello visión artificial.
- Comprobar la aplicación desarrollada con Processing para evaluar la precisión del robot, implementado el empaque de pelotas de tenis de mesa en sus respectivos envases.

CAPÍTULO 2

FUNDAMENTACIÓN TEÓRICA

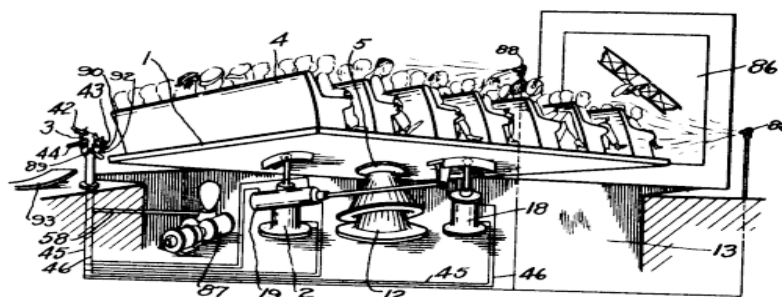
2.1 Robots Paralelos

Un robot de cadena cerrada o manipulador paralelo, también llamado robot de cinemática paralela consta en su estructura de una plataforma movable que se encuentra a una base inmóvil o fija unida a varios brazos, la mayoría de veces cada brazo es controlado por un actuador. (Isidro Zabalza, 2007)

Las ventajas que brinda este tipo de estructuras comparadas con los robots serie son velocidades y aceleraciones altas, poseen precisión elevada debido a su alta rigidez ya que el peso es repartido entre sus articulaciones, además de manejar cargas mayores, si bien es cierto que cuenta con estas ventajas también poseen mayores requerimientos como: cinemáticas más complejas, mayores exigencias en los algoritmos de control debido a la alta dinámica que poseen estas estructuras. (V. Poppeová, 2011)

Según (Isidro Zabalza) comenta que “los modelos matemáticos usados hace más de 50 años han avanzado de la misma manera que los métodos de simulación y corrección de errores, es por eso que desde el modelo diseñado por J.E. Gwinnett en 1931 como se muestra en la Figura 2.1 y este evolucionando en diversos modelos en el presente, este tipo de robots ha incursionado en múltiples y variadas aplicaciones de robots trepadores, sistemas de posicionamiento de objetos como: antenas, también en sistemas paralelos de tele operación en medicina, manufactura, entre otros.” (pág. 2)

Figura 2.1 Primer Robot Paralelo



Grafica del Primer Robot Paralelo del que se tiene conocimiento el cual es una plataforma móvil de cine. (Isidro Zabalza, 2007)

En la industria hay dos estructuras de tipo paralelo que tiene una participación sobresaliente debido a sus cualidades el primero en el cual nos centraremos en un estudio profundo más adelante es el robot Delta que vio la luz en el año de 1989, propuesto por Reymond Clavel, un robot con tres traslaciones o tres grados de libertad que por sus tres motores fijos en una plataforma inmóvil puede alcanzar una aceleración de 50 veces la gravedad, una cualidad inigualable por cualquier robot tipo serie, el segundo el Robot Hexápodo como se muestra en la Figura 2.2 con la cualidad de poseer seis grados de libertad, tres de traslación y tres de rotación. (López-Betancourt-Martínez-Ancira-Ramirez, 2011)

Figura 2.2 Robot Hexápodo



Grafica de un Robot Hexápodo con 6 grados de libertad desarrollado por la Universidad Autónoma de Nuevo León. (López-Betancourt-Martínez-Ancira-Ramirez, 2011)

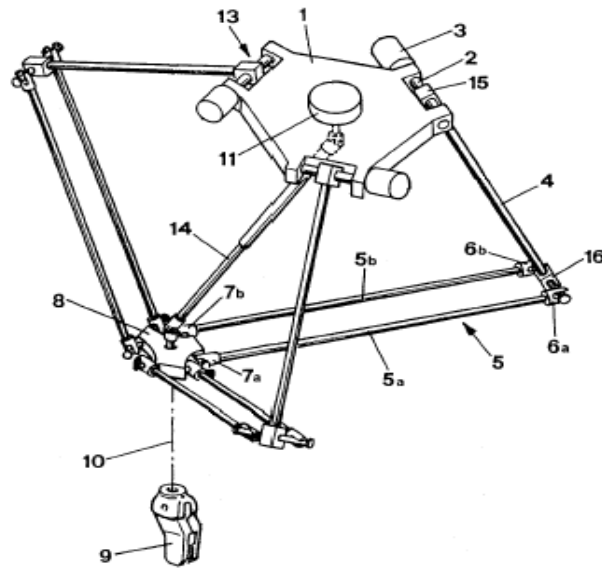
2.1.1 Robot Delta

Reymond Clavel en la década de los 80 propuso un modelo de robot que luego estaría dentro de los más exitosos de la década, la idea mentalizada era de proponer un diseño en el cual tres articulaciones móviles estén fijas a una base inmóvil usando un paralelogramo para cada una de ellas, otorgándole a cada articulación tres enlaces de entrada uno por cada uno de los paralelogramos utilizados en cada miembro articulado, estos enlaces se encuentran montados sobre palancas de rotación a través de juntas de revolución, las juntas de revolución de las palancas giratorias cuentan un doble accionar: el primero usa servo motores (motores de rotación AC o DC), el segundo usa actuadores lineales. (Zsombor-Murray, 2004)

La idea básica del robot delta es una base fija de un triángulo equilátero que posee tres eslabonamientos similares y simétricos fijos a la base dándole tres grados de libertad,

por último, para transmitir el movimiento de rotación desde la base hacia un efector en el extremo montado en la base móvil dándole un grado de libertad si fuese el caso dependiendo de la tarea que va a realizar, la Figura 2.3 muestra una figura Esquemática del Robot Delta patentada en 1991 por Raymond Clavel. (V. Poppeová, 2011)

Figura 2.3 Robot Delta



Grafica Esquemática del Robot Delta presentado por Raymond Clavel (V. Poppeová, 2011)

La relación que presenta entre carga y potencia es alta ya que los accionamientos de potencia conectan directamente la base inmóvil del robot al efector móvil final, sirviendo de elementos estructurales que actúan en el mismo instante, permitiéndoles manipular cargas mayores a su propio peso entonces el robot delta hace uso de sus actuadores montados en la base móvil, con sus enlaces que de baja masa lo hace el candidato perfecto para diversas tareas de colocación y recogida de objetos livianos (peso aproximado de 10 gr a 1,5 kg), alcanzando aceleraciones de 12 G en aplicaciones industriales y 50 G en entornos experimentales. (V. Poppeová, 2011)

2.1.2 Actuadores finales

Existen diferentes tareas que puede desempeñar el robot delta con sus respectivos actuadores, sin embargo, los más usadas son las pinzas y estas se dividen respecto a su sistema de sujeción y son:

2.1.2.1 Ventosa de Vacío

La ventosa de vacío está construida de goma inflable que al contacto con un material despliega un toque suave de aire haciendo crecer un diámetro tanto como 1,86 veces el diámetro desinflado insertando presión dentro de la vejiga haciendo actuar una fuerza de retención que se distribuye sobre una superficie mayor dando como resultado una carga más alta con un agarre más firme y una mayor capacidad de carga. El toque suave de la goma y el aire permite manipular desde vidrio hasta objetos de plástico sin dañar su superficie. (Ralph-Christoph Weber, 2011)

Figura 2.4 Ventosa de Vacío

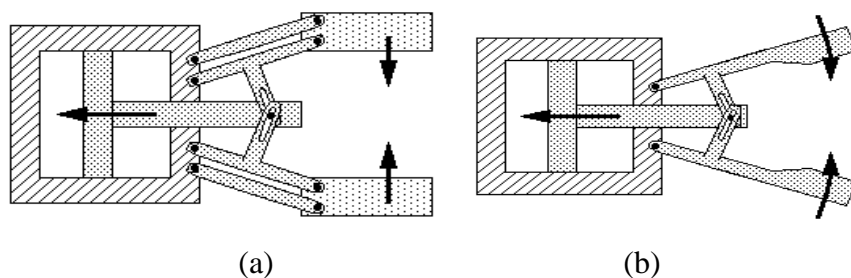


Diferentes ventosas de vacío extraídas del manual de operación festo pneumatics
(Ralph-Christoph Weber, 2011)

2.1.2.2 Pinza de Presión o Gripper

El tipo de pinza de presión más común es denominado pivotante Figura 2.5 caso (a), el funcionamiento de la pinza permite abrir los dedos que rotan y están en relación con los sitios asegurados del pivote. Hay otro tipo de pinza llamada pinza de traslación lineal, la Figura 2.5 caso (b). Los dedos se abren y se cierran dando lugar un movimiento semejante entre sí. (Soriano, 2015)

Figura 2.5 Pinza de Presión O Gripper



Grafica de pinza de presión o gripper. (a) Caso de pinza pivotante (b) Caso de pinza con movimiento
(Soriano, 2015).

2.1.2.3 Electroimán

El accionamiento eléctrico de este actuador permite la recolección de elementos metálicos gracias al campo magnético que genera el electroimán colocado en la punta de la pinza. La elección de este actuador solo permite recoger objetos con propiedades ferromagnéticas. (Goudsmith, 2013)

Figura 2.6 Electroimán



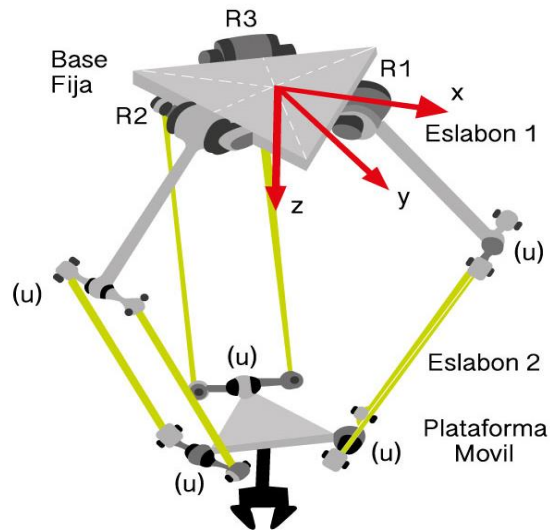
Electroimán extraído del manual de Goudsmith-Systems. (Goudsmith, 2013).

2.2 Cinemática de un Robot Paralelo

2.2.1 Modelo Cinemático General de un Robot Delta

Antes de definir el modelo cinemático del robot delta, se debe enfocar primero en el estudio de su geometría, como se observa en la Figura 2.7, (Williams, 2016) indica que el “Robot Delta cuenta con una base inmóvil o fija, en la cual se ubican tres servomotores (S1, S2, S3) separados a 120 grados entre sí y a una misma distancia medida desde el centro, una base móvil que se conecta mediante tres cadenas cinemáticas las cuales están constituidas por dos eslabones, unidos por juntas universales (U), existe una segunda plataforma móvil que se une a la primera y cuenta con un servomotor S4, para permitir la orientación de una pinza o gripper”.

Figura 2.7 Estructura del Robot Paralelo



Descriptive Geometric Kinematic Analysis of Clavel's "Delta" Robot P.J. (Williams, 2016)

$$F = \lambda(n - j - 1) + \sum_{i=1}^j f_i \quad \text{Ec. (2.1)}$$

La Ecuación 2.1 descrita anteriormente nos permite los grados de libertad del prototipo del robot Delta. (Williams, 2016)

Donde:

F= Grados de libertad del mecanismo

λ = Grados de libertad del espacio en el cual el mecanismo funciona

n= número de cuerpos rígidos del mecanismo

j= número de articulaciones

f_i = grados de la articulación i

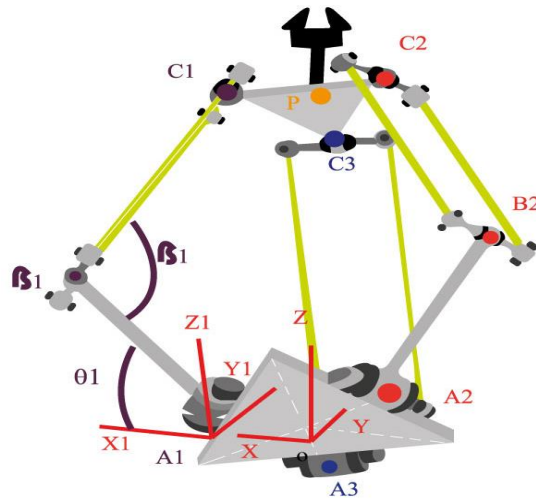
por lo tanto, el robot delta tiene los siguientes grados de libertad:

$$F = 6(9 - 10 - 1) + 16 = 4$$

Además (Williams, 2016) indica que “el robot delta es un mecanismo espacial, es por eso que se ha utilizado el valor de $\lambda=6$, formado por 6 eslabones una plataforma inmóvil y dos móviles $n=9$, seis articulaciones universales, cuatro de revolución $j=10$, las articulaciones universales conformadas con dos grados de libertad”.

Las tres cadenas cinemáticas ($i=1, 2, 3$) formadas por dos eslabones cada una $L1$ y $L2$ cómo se puede apreciar en la Figura 2.8.

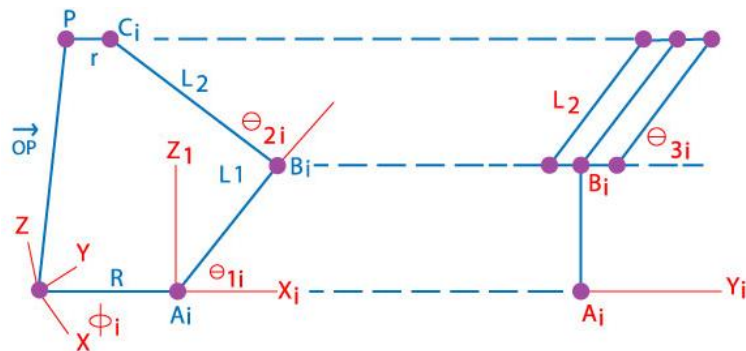
Figura 2.8 Robot Delta y su modelo geométrico



Descriptive Kinematic Analysis of Clavel's "Delta" Robot P.J. (Williams, 2016)

La Figura 2.8. donde se explica que: “el vector \overrightarrow{OP} es el vector de posicionamiento del centro de la estructura móvil, existe también un ángulo denominado θ_{1i} y este es el ángulo que se forma entre X_{i1} y el vector $\overrightarrow{A_i B_i}$.” (Vivas, 2013)

Figura 2.9 ángulos desde su: izq) Perspectiva frontal, der) Perspectiva lateral



Ángulos formados por las articulaciones (Vivas, 2013)

Para conocer las coordenadas del vector respecto a un sistema de referencia $O'X'Y'Z$ se usará la siguiente matriz de transformación T en la Ecuación 2.2. (Vivas, 2013)

$$\vec{r} = [r_x, r_y, r_z]^T \text{ de acuerdo al vector } \vec{r}' = [r_x', r_y', r_z']^T.$$

$$\begin{bmatrix} r_x \\ r_y \\ r_z \\ 1 \end{bmatrix} = T \begin{bmatrix} r_x' \\ r_y' \\ r_z' \\ 1 \end{bmatrix} \quad \text{Ec. (2.2)}$$

Donde la matriz T se muestra en la Ecuación 2.3. (Vivas, 2013)

$$T = \begin{bmatrix} R_{3 \times 3} & P_{3 \times 1} \\ 0_{1 \times 3} & 1 \end{bmatrix} = \begin{bmatrix} \text{Rotacion} & \text{Traslacion} \\ 0_{1 \times 3} & 1 \end{bmatrix} \quad \text{Ec. (2.3)}$$

La matriz transpuesta del sistema A_i, X_i, Y_i, Z_i es la siguiente.

$$T(P, (Z, \alpha)) = \begin{bmatrix} \cos(\alpha) & -\sin(\alpha) & 0 & x\cos(\alpha) - y\sin(\alpha) \\ \sin(\alpha) & \cos(\alpha) & 0 & x\sin(\alpha) + y\cos(\alpha) \\ 0 & 0 & 1 & z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{Ec. (2.4)}$$

Aplicando la Ecuación 2.1 con la transformación homogénea en la Ecuación 2.2 se obtienen los vectores $\overrightarrow{OA_l}, \overrightarrow{OB_l}$ y $\overrightarrow{OC_l}$. (Vivas, 2013)

$$\overrightarrow{OA_l} = \begin{bmatrix} \cos(\phi_i) & -\sin(\phi_i) & 0 & R\cos(\phi_i) \\ \sin(\phi_i) & \cos(\phi_i) & 0 & R\sin(\phi_i) \\ 0 & 0 & 1 & z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} R\cos(\phi_i) \\ R\sin(\phi_i) \\ 0 \\ 1 \end{bmatrix} \quad \text{Ec. (2.5)}$$

$$\begin{aligned} \overrightarrow{OB_l} &= \begin{bmatrix} \cos(\phi_i) & -\sin(\phi_i) & 0 & R\cos(\phi_i) \\ \sin(\phi_i) & \cos(\phi_i) & 0 & R\sin(\phi_i) \\ 0 & 0 & 1 & z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} R\cos(\phi_i) \\ R\sin(\phi_i) \\ 0 \\ 1 \end{bmatrix} = \\ &= \begin{bmatrix} \cos(\phi_i) + (R + L_1\cos(\phi_{li})) \\ \sin(\phi_i) + (R + L_1\sin(\phi_{li})) \\ L_1\sin(\phi_{li}) \\ 1 \end{bmatrix} \quad \text{Ec. (2.6)} \end{aligned}$$

$$\begin{aligned}
\overrightarrow{OC_l} &= \begin{bmatrix} \cos(\phi_i) & -\sin(\phi_i) & 0 & R\cos(\phi_i) \\ \sin(\phi_i) & \cos(\phi_i) & 0 & R\sin(\phi_i) \\ 0 & 0 & 1 & z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} P_x\cos(\phi_i) + P_y\sin(\phi_i) + r - R \\ P_y\cos(\phi_i) - P_x\sin(\phi_i) \\ P_z \\ 1 \end{bmatrix} = \\
&= \begin{bmatrix} P_x + r\cos(\phi_i) \\ P_y + r\sin(\phi_i) \\ P_z \\ 1 \end{bmatrix} \quad \text{Ec. (2.7)}
\end{aligned}$$

Adicionalmente, se puede encontrar los vectores $\overrightarrow{B_lA_l}$, $\overrightarrow{B_lC_l}$ y $\overrightarrow{A_lC_l}$, como se indica en la Ecuación 2.8. (Vivas, 2013)

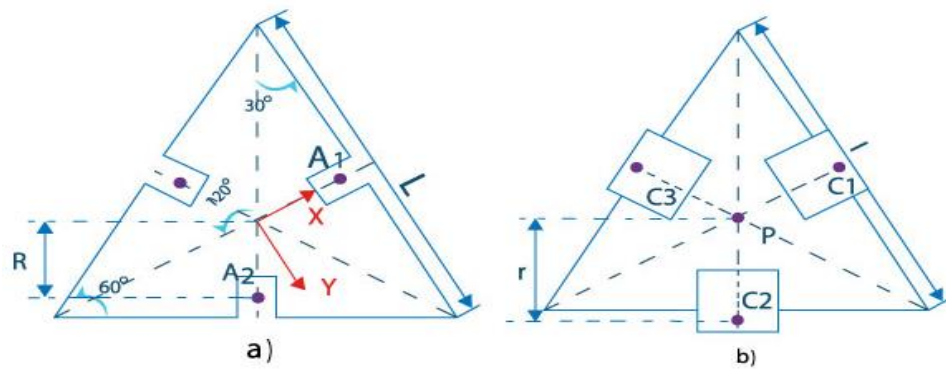
$$\overrightarrow{B_lA_l} = \overrightarrow{OA_l} - \overrightarrow{OB_l} = \begin{bmatrix} -L_1 \cos(\phi_i) \cos(\phi_{1i}) \\ -L_1 \sin(\phi_i) \cos(\phi_{1i}) \\ -L_1 \sin(\phi_{1i}) \end{bmatrix} \quad \text{Ec. (2.8)}$$

$$\overrightarrow{B_lC_l} = \overrightarrow{OC_l} - \overrightarrow{OB_l} = \begin{bmatrix} P_x - \cos(\phi_i)(R + L_1 \cos(\phi_{1i}) - r) \\ P_y - \sin(\phi_i)(R + L_1 \cos(\phi_{1i}) - r) \\ P_z - L_1 \sin(\phi_{1i}) \end{bmatrix} \quad \text{Ec. (2.9)}$$

$$\overrightarrow{A_lC_l} = \overrightarrow{OC_l} - \overrightarrow{OA_l} = \begin{bmatrix} P_x + \cos(\phi_i)(r - R) \\ P_y + r\sin(\phi_i)(r - R) \\ P_z \end{bmatrix} \quad \text{Ec. (2.10)}$$

Por facilidad de notación, en las Ecuación 2.1 a la Ecuación 2.10 han sido reducidas a su mínima expresión.

La Figura 2.10 detalla la representación de: a) el armazón fijo, b) el armazón móvil



Detalle de las bases de la plataforma Delta (Williams, 2016),

2.2.2 Cinemática Inversa

La cinemática inversa es un método por el cual se puede conocer la posición de los miembros actuadores de acuerdo a las constantes que interactúan con los miembros del robot, si se conoce la posición final del miembro actuador, para este caso se conoce la posición final del armazón móvil del robot, pudiendo enviar ordenes desde un controlador para que se muevan hasta el lugar deseado. (Zsombor-Murray, 2004)

Figura 2.11 Representación vectorial de la primera cadena cinemática

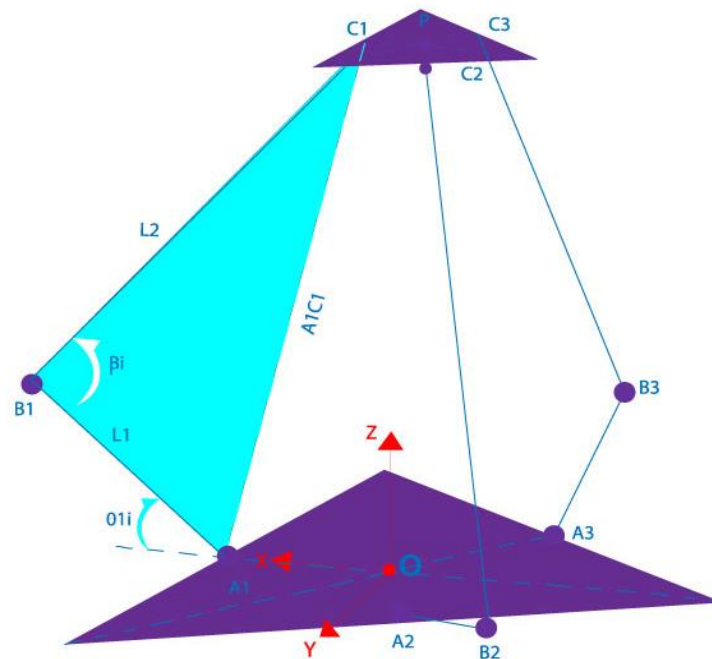


Diagrama Vectorial de la plataforma Delta (Zsombor-Murray, 2004)

$A_i B_i C_i$ como se aprecia en la Fig.2.11 es un triángulo, $A_i C_i$ es calculado mediante identidad trigonométrica, como se indica en la Ecuación 2.11. (Zsombor-Murray, 2004)

$$A_i L_i^2 = L_1^2 + L_2^2 - 2L_1 L_2 \cos(\beta_i) \quad \text{Ec. (2.11)}$$

En la Ecuación 2.11, el $\cos(\beta_i)$ es una variable sin calcular, obtenida mediante producto escalar. (Zsombor-Murray, 2004)

$$\overrightarrow{B_i A_i} * \overrightarrow{B_i C_i} = (B_i A_i)(B_i C_i) \cos(\beta_i) \quad \text{Ec. (2.12)}$$

Como $B_i A_i = L_1$ y $B_i C_i = L_2$ entonces, se procede a resolver, como se indica en la Ecuación 2.13. (Zsombor-Murray, 2004)

$$\cos(\beta_i) = \frac{\overrightarrow{B_i A_i} * \overrightarrow{B_i C_i}}{L_1 L_2} \quad \text{Ec. (2.13)}$$

Remplazando la ecuación 2.13 en la Ecuación 2.11 se obtiene la Ecuación 2.14. (Zsombor-Murray, 2004)

$$A_i C_i^2 = L_1^2 + L_2^2 - B_i A_i * B_i C_i \quad \text{Ec. (2.14)}$$

Y se procede a calcular la solución como se muestra en la Ecuación 2.15. (Zsombor-Murray, 2004)

$$2 \overrightarrow{B_i A_i} * \overrightarrow{B_i C_i} = 2(L_1^2 - L_1 C_{\theta i} (P_x C_{\phi i} + P_y S_{\phi i} + r - R) - L_1 S_{\theta i} P_z) \quad \text{Ec. (2.15)}$$

Remplazando la Ecuación 2.15 en la Ecuación 2.14 se obtiene la Ecuación 2.16. (Zsombor-Murray, 2004)

$$A_i C_i^2 = L_1^2 - L_2^2 + 2L_1 C_{\theta i} (P_x C_{\phi i} + P_y S_{\phi i} + r - R) + 2L_1 S_{\theta i} P_z \quad \text{Ec. (2.16)}$$

La magnitud del vector de la Ecuación (2.10) $\overrightarrow{AC_1}$ es:

$$A_i C_i^2 = (P_x + C_{\theta i}(r - R))^2 + (P_y + S_{\theta i}(r - R))^2 + P_z^2 \quad \text{Ec. (2.17)}$$

Igualando Ecuación 2.17 y la Ecuación 2.16 se obtiene la Ecuación 2.18 de la siguiente forma. (Zsombor-Murray, 2004)

$$a_i S_{\theta_{li}} + b_i C_{\theta_{li}} = C_i \quad \text{Ec. (2.18)}$$

Donde:

$$ca_i = 2P_z L_1$$

$$cb_i = 2L_1 (P_x C_{\theta_i} + P_y S_{\theta_i} + r - R)$$

$$cc_i = (P_x + C_{\theta_i}(r - R)^2 + (P_y + S_{\theta_i}(r - R)^2) + P_z^2 + L_1^2 + L_2^2$$

Es posible describir la Ecuación 2.18 mediante fórmulas trigonométricas. (Zsombor-Murray, 2004)

$$\theta_{li} = \sin^{-1} \left(\frac{cc_i}{\sqrt{ca_i^2 + cb_i^2}} \right) - \tan^{-1} \left(\frac{cb_i}{ca_i} \right) \quad \text{Ec. (2.19)}$$

El ángulo θ_{11} , $i=1$, $\theta_1 = 0^\circ$, por lo tanto:

$$ca_1 = 2P_z L_1$$

$$cb_1 = 2L_1 (r + P_x - R)$$

$$\theta_{11} = \sin^{-1} \left(\frac{cc_i}{\sqrt{ca_i^2 + cb_i^2}} \right) - \tan^{-1} \left(\frac{cb_1}{ca_1} \right) \quad \text{Ec. (2.20)}$$

El ángulo θ_{12} , $i=2$, $\theta_2 = 120^\circ$, entonces:

$$ca_2 = 2P_z L_1$$

Simplificando se obtiene la Ecuación 2.21. (Zsombor-Murray, 2004)

$$\theta_{12} = \sin^{-1} \left(\frac{cc_2}{\sqrt{ca_2^2 + cb_2^2}} \right) - \tan^{-1} \left(\frac{cb_2}{ca_2} \right) \quad \text{Ec. (2.21)}$$

El ángulo θ_{13} es encontrado de la siguiente manera: i adopta valores de los ángulos formados entre sí por lo cual θ_3 es de 240° , entonces:

$$ca_3 = 2P_z L_1$$

Utilizando despeje de fórmulas y simplificación algebraica tenemos la Ecuación

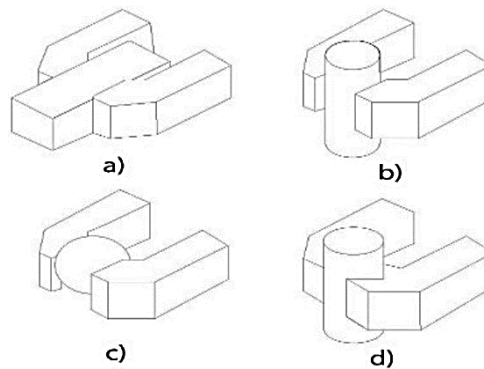
2.22. (Zsombor-Murray, 2004)

$$\theta_{13} = \sin^{-1} \left(\frac{cc_3}{\sqrt{ca_3^2 + cb_3^2}} \right) - \tan^{-1} \left(\frac{cb_3}{ca_3} \right) \quad \text{Ec. (2.22)}$$

2.3 Cinemática general de una Pinza o Gripper

Es importante tener en cuenta aquellas partes de la pieza por donde se podrá llevar a cabo la operación de pinzado ya que la sujeción externa generalmente precisa más espacio alrededor de la pinza. (Soriano) indica que “las cargas puntuales debidas al proceso de sujeción de muchas piezas, debido a determinadas características superficiales pueden dañarse por aquellas pinzas cuya sujeción suponga la aplicación de una carga puntual”, estos tipos de carga puntual producidos en el proceso de sujeción de un objeto pueden dividirse en cuatro grupos claramente diferenciados en la Figura 2.12. que se detallada a continuación:

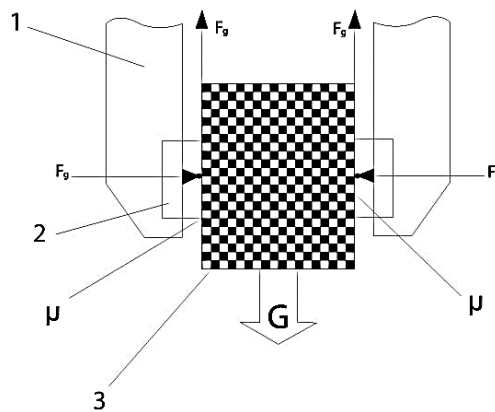
Figura 2.12 Tipos de carga puntual debido a la sujeción



Cargas puntuales referentes de sujeciones de piezas (Soriano, 2015)

Según el autor (Pose) indica que: “en el punto de contacto se produce una deformación, según la orientación de la pinza, originado por un factor externo llámese a este factor o coeficiente de rozamiento μ ” (págs. 42-43).

Figura 2.13 Función simple de Sujeción



Elementos que intervienen en la sujeción de una pieza (Pose, 2015)

- 1- Dedo
- 2- Mandíbula
- 3- Pinza
- 4- μ = coeficiente de rozamiento

$$Fg = \frac{m \cdot G}{\mu \cdot n} \quad \text{Ec. (2.23)}$$

En donde las variables de la fórmula utilizada son las siguientes. (Pose, 2015)

Fg = Fuerza de sujeción en N wtons

G = Fuerza del objeto en N wtons

g = Aceleraci n de g en $\frac{m}{s^2}$

m = Masa en kg

n =Numero de mand bulas o pinzas

μ = Coeficiente de rozamiento

Como nos indica (Pose) , “en la f rmula se considera el n mero de dedos ya que se crea una fuerza de rozamiento F_r en cada mand bula. Con tres puntos de contacto, $n=3$, La Figura 2.14. muestra una pinza de tres dedos y una soluci n basada en una pinza de dos dedos. En este  ltimo caso, la fuerza de pinzado se reparte en la mand bula V en fuerzas de contacto F_r ”. (p g. 51)

Figura 2.14 Representación gráfica de las fuerzas de contacto para mandíbulas.

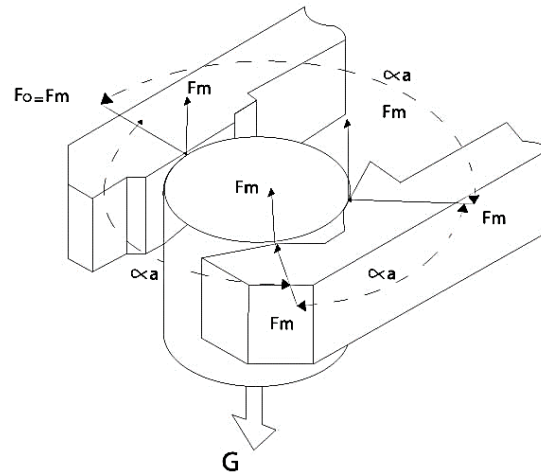


Diagrama de fuerzas involucradas en el contacto o agarre de piezas (Pose, 2015)

Además (Pose) indica que “el movimiento de descenso, la fuerza de inercia actúa oponiéndose a la fuerza debida al peso y hace que la pieza sea realmente “más ligera” Sin embargo, durante la fase de frenado se produce una fuerza de deceleración. En el caso de movimientos laterales, la fuerza de la inercia es absorbida por la forma de las mandíbulas, lo que no exige fuerzas de rozamiento adicionales” (pág. 54)

Las Fuerzas de inercia pueden calcularse, en general, como indica la Ecuación 2.24. (Pose, 2015)

$$Fb = m \cdot a \quad \text{Ec. (2.24)}$$

O en el caso de movimientos rotativos la Ecuación 2.25.

$$Fb = m \cdot r \cdot \varepsilon \quad \text{Ec. (2.25)}$$

Términos utilizados en las fórmulas citadas

a = Aceleración lineal en $\frac{m}{s^2}$

n =Masa del objeto manipulado en Kg

r = Distancia al punto de giro en m

ε = Aceleración angular en $\frac{rad}{s^{-1}}$

A continuación, se ilustra un ejemplo de cálculo de fuerzas de sujeción en condiciones de elevación, recorrido lateral y paro de emergencia, el peso oficial de una pelota de tenis de mesa es 2.7g. Cabe destacar que los valores obtenidos en este ejemplo sólo sirven para la adquisición del servo motor que mejor se adapte al valor requerido. (Pose, 2015), (Ralph-Christoph Weber, 2011)

Fuerza de sujeción durante la elevación Ecuación 2.26. (Ralph-Christoph Weber, 2011)

$$F_g = \frac{m(g+a)}{\mu \cdot n} = \frac{0.0027 \cdot (9.81+5)}{0,2 \cdot 2} = 0.133N \quad \text{Ec. (2.26)}$$

Fuerza de sujeción durante el recorrido lateral Ecuación 2.27. (Ralph-Christoph Weber, 2011)

$$F_g = \frac{m \cdot g}{\mu \cdot n} + m \cdot a = \frac{0.0027 \cdot 9.81}{0,2 \cdot 2} + 0.0027 \cdot 5 = 5.069N \quad \text{Ec. (2.27)}$$

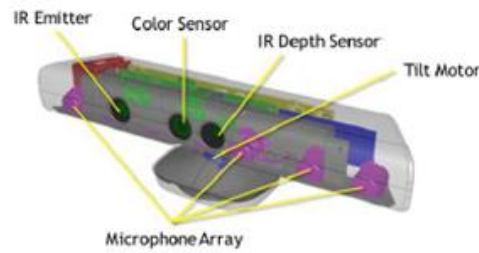
Fuerza de sujeción hacia abajo durante el Paro de emergencia (después del paro de emergencia), Ecuación 2.28. (Ralph-Christoph Weber, 2011)

$$F_g = \frac{m(g+an)}{\mu \cdot n} = \frac{0.0027 \cdot (9.81+10)}{0,2 \cdot 2} = 0.134N \quad \text{Ec. (2.28)}$$

2.4 Sensor Kinect

Microsoft para adentrarse en el mercado de los videojuegos, presenta una versatilidad en la jugabilidad, desarrolla un Sensor de profundidad para capturar movimientos del usuario, que no solo es usado en el ámbito para el cual fue diseñado, se lo emplea en entornos de control de dispositivos, educativos, video conferencias, etc. El sensor Kinect es prácticamente una barra horizontal con 23 cm de largo, que está conectada a una base móvil circular. (Ojosnegros Ramos, 2014), (Flacco Fabrizio, 2017)

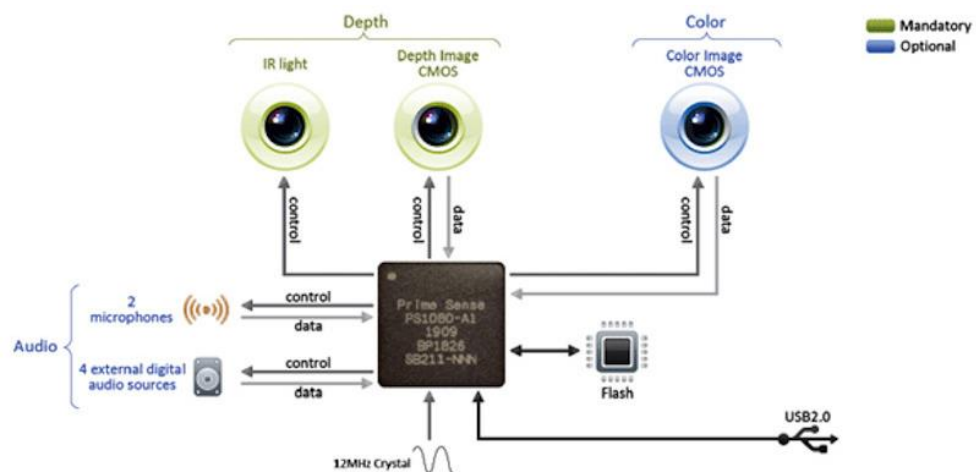
Figura 2.15 Secciones principales del Sensor Kinect



Partes principales del Sensor Kinect (Ojosnegros Ramos, 2014)

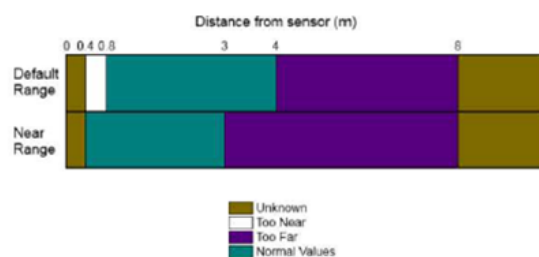
El sensor de profundidad puede ver espacios en 3D en casi cualquier lugar y e todo tipo de condición ambiental, debido a la alta captura de movimientos presentes en el sensor utilizando un sensor de infrarrojos. (Doumanoglou, 2013)

Figura 2.16 Composición interna del Sensor Kinect



Componentes internos del Sensor Kinect V1, (Ojosnegros Ramos, 2014)

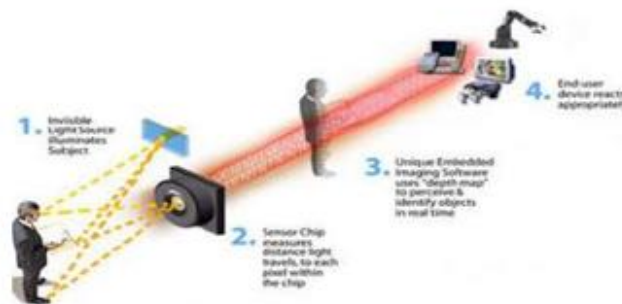
Figura 2.17 Distancias de detección modo por defecto y modo near



Rango de captura de movimientos del Sensor Kinect
(Ojosnegros Ramos, 2014)

También (Ojosnegros Ramos) muestra que “los movimientos de los objetos son interpretados por el procesador que son capturados por la cámara del Kinect en movimientos son mostrados en pantalla como se muestran en la Figura 2.18., estos movimientos son capturados y luego con el algoritmo son contextualizados y ejecutados, por ejemplo, si se está usando el sensor en un video juego en donde se necesite agacharse o mover las manos, el algoritmo busca la identificación de estos en tiempo real para mostrarlos en pantalla.” (pág. 9)

Figura 2.18 Descripción de captura de movimientos del Sensor Kinect

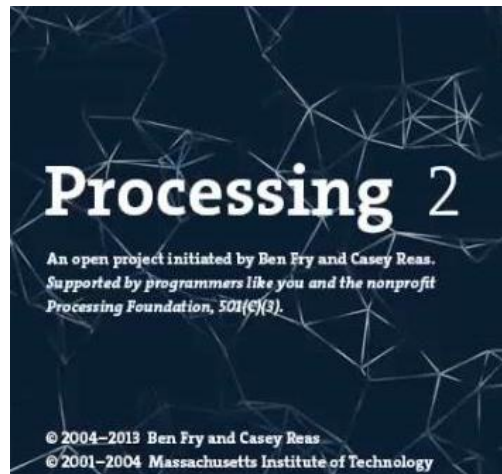


Método de captura de movimientos utilizados por el sensor Kinect V1 (Ojosnegros Ramos, 2014)

2.5 Processing

Ben Fry junto con su colega Casey desarrollo un programa de código libre que se lo utiliza para aplicaciones de identificación de imágenes, animaciones e interacciones. Este entorno de programación en código abierto fue desarrollado inicialmente para servir herramienta de aprendizaje para diseñadores gráficos, también se ha convertido en una herramienta para la generación de trabajos profesionales, estudiantiles, artísticos, desarrolladores, investigadores y aficionados hacen uso de Processing para el aprendizaje, creación y producción. El código que se escribe en java se compila en bytecode que se ejecuta luego por Java Virtual Machine que reside en su ordenador, esto hace que el programador escriba el software sin la preocupación de un sistema operativo que se ejecute en él. (Ben Fry, 2015)

Figura 2.19 Logo de inicio de software Processing 2.2.1



Logo de entrada al entorno de Processing 2 (Ben Fry, 2015)

Processing fue creado por Ben Fry y Casey Reas en la primavera de 2001, como estudiantes de postgrado en el MIT Media Lab. De la dirección de Jhon Maeda y el grupo de investigación en computación. El desarrollo continuo, mientras que Cassey continuo con su carrera Artística y de docencia y Ben siguió un doctorado fundando Fathon Information Design. (Ben Fry, 2015)

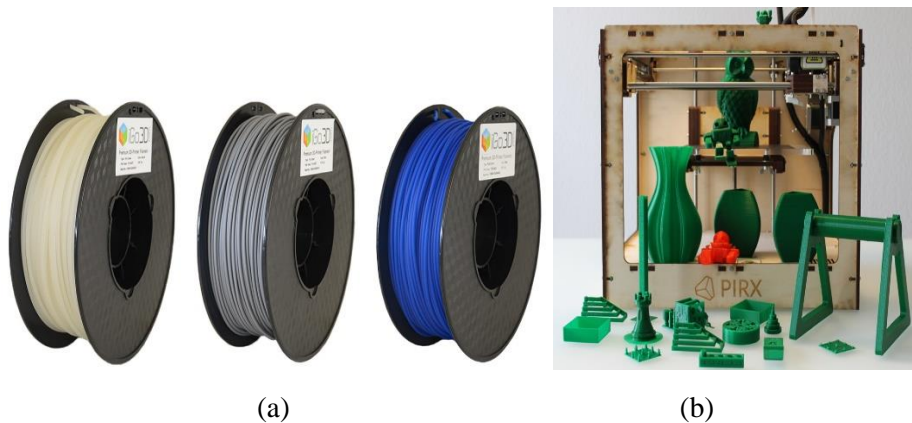
2.6 Polímeros de acrilonitrilo-butadieno-estireno (plástico ABS)

Compuestos principalmente de tres unidades monoméricas: acrilonitrilo, butadieno y estireno. Plástico tiene muchas propiedades versátiles que incluyen resistencia térmica, peso ligero, fácil capacidad de conformado, la reflectividad etc. En cuanto a todas estas propiedades que ha abierto nueva forma de uso para el material como ABS / PC. (Vishwakarma, 2017)

Debido a la combinación de las propiedades de los componentes individuales, mezclas de polímeros proporcionan un precio-servicio-relación óptima debido a sus características rendimientos superiores a precios razonables. La introducción de poros para el sistema material, ayuda en la reducción adicional del peso y la absorción de energía a través de una gran deformación a la compresión, y esto se (fabricación de estructuras porosas) más fácil y más rentable en comparación con los metales y cerámicas. (Vishwakarma, 2017)

“El polímero de acrilonitrilo-butadieno-estireno tiene muchas propiedades que incluyen peso ligero, fácil conformabilidad, resistencia a la abrasión, etc. Esto es útil para la aplicación industrial, haciendo cubiertas decorativas, ruedas, piezas del aire acondicionado, la metalización de plástico sirve para hacer impresiones 3D que lo hará ser un material exigente en un futuro próximo” como hace referencia (Vishwakarma). (págs. 13-16)

Figura 2.20 Plástico ABS presente en el mercado



(a) Imagen perteneciente al plástico ABS en su forma comercial.

(b) plástico ABS utilizado para

el método de impresión 3D (Vishwakarma, 2017)

CAPÍTULO 3

DESARROLLO Y CONSTRUCCIÓN DEL HARDWARE Y DEL SOFTWARE

3.1 Visión General

En el presente Capitulo se presentará el diseño del modelo del robot Delta, empezando con el modelado de cada una de las piezas por separado para posteriormente agruparlas utilizando el entorno Autocad 2015, determinando las piezas necesarias para la construcción del robot, características del material y para concluir con su implementación física. (Referirse al Anexo A para una información detallada acerca de las piezas construidas en esta sección),

3.2 Hardware Mecánico

Es parte fundamental de la plataforma delta el modelado mecánico el cual debe mostrar características robustas junto a un óptimo funcionamiento para realizar las pruebas de control, para el diseño del robot este se utilizará ingeniería inversa aplicada al modelo Delta LLC Army que se puede apreciar en la figura 3.1.

Figura 3.1 Modelos de diferentes robots delta en el mercado



De Izq a Der: Delta Junky, Robot Army, Tapsterbot (Rick Waldron, 2015).

3.2.1 Diseño de Piezas en Autocad 2015

Se determinó que es imperativo empezar por la parte del modelado de las piezas del robot para su posterior implementación, por tal motivo se procederá a realizar el modelado individual de las piezas elaboradas en el entorno Autocad 2015, aplicando ingeniería inversa al Robot Army llc, para su posterior construcción e implementación.

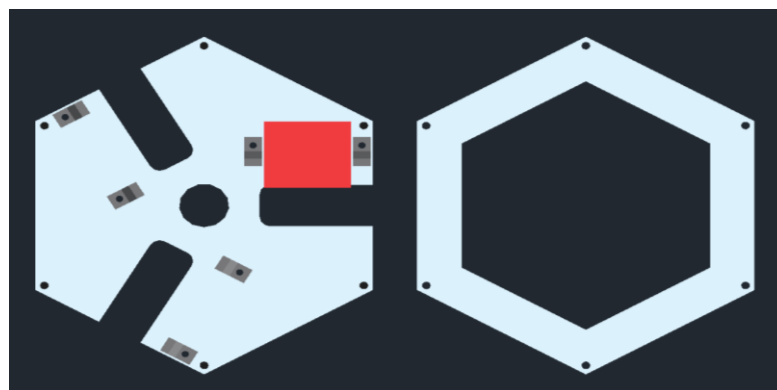
3.2.2 Base Fija

La base fija es una parte mecánica del robot donde se acoplan los elementos mecánicos del robot tales como: servo motores junto con sus brazos, por tal motivo debe cumplir con la característica de ser robusto e inmóvil de tal forma que las diferentes partes mecánicas cumplan con el propósito de mover el efector sin tambalearse o descuadrarse haciendo posible un correcto posicionamiento del gripper o pinza.

3.2.2.1 Diseño de la base fija

Para el Diseño de la base fija se tomó en cuenta el espacio ocupado por los tres servomotores que comandan a los brazos mecánicos del robot Delta, también se tomó en cuenta el desplazamiento de cada uno de ellos y que se encuentren distribuidos correctamente. El diseño de la base móvil se lo realizo en el software AutoCad 2015 en la Figura 3.2. se denota el diseño de la base móvil de una perspectiva frontal y superior con sus respectivas medidas.

Figura 3.2 Diseño de la base Fija del Robot Delta

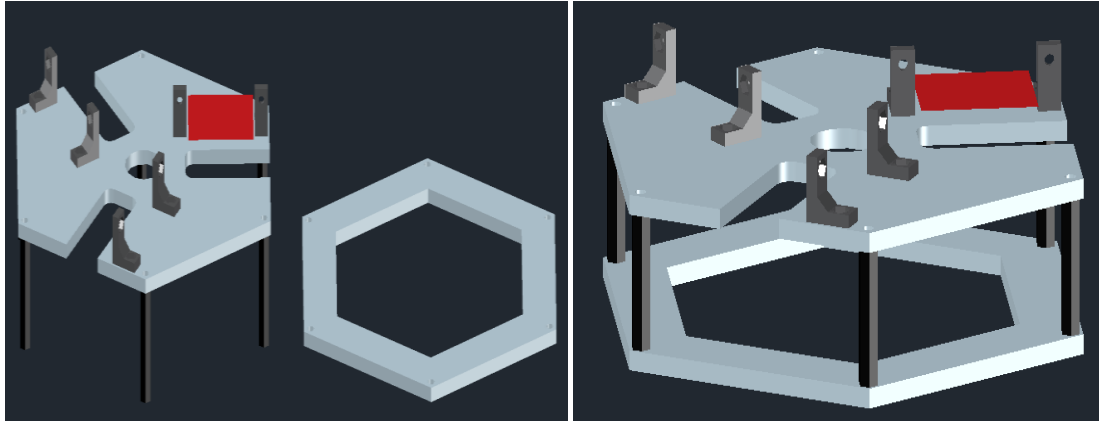


Vista Superior del diseño de la base fija del Robot. Elaborado por: Esteban Martínez

En la Figura 3.3. se denota dos piezas, (a) hace referencia a la base fija #1 que soporta las partes mecánicas de todo el robot servos, brazos, efectores, etc. (b) hace referencia

a la base fija #2 que estará anclada a cualquier tipo de soporte. Los dos se acoplan entre sí como se aprecia en la Figura 3.3 (b). mediante a barras de metal.

Figura 3.3 Base Fija del Robot Delta acoplamiento



(a)

(a) Vista en perspectiva de las bases fijas

(b)

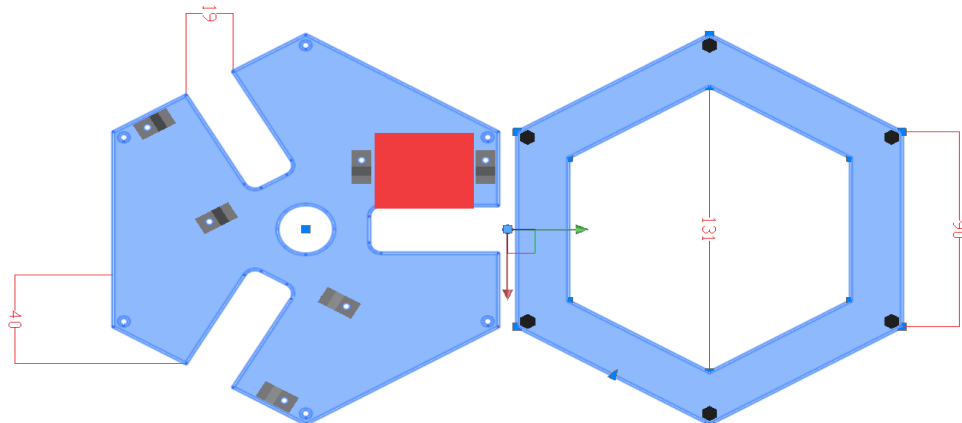
(b) Bases fijas totalmente Acopladas.

Elaborado por: Esteban Martínez

3.2.2.2 Construcción de la base fija

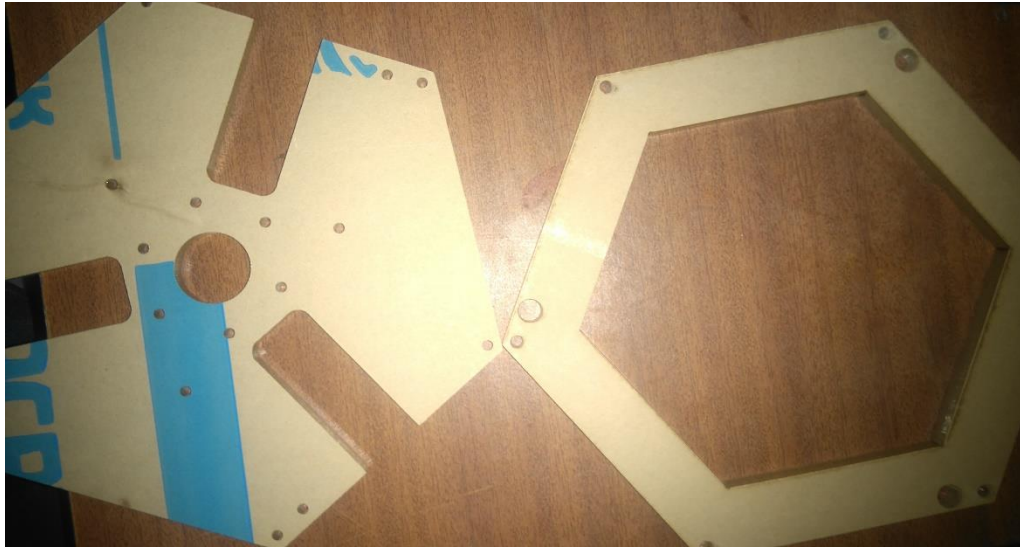
Para la construcción de la carcasa se cortó en acrílico mediante laser, como se muestra en la Figura 3.5, no se optó por la opción de impresión en esta parte del diseño debido a que no son piezas que necesiten de un acabado detallado y perfiles sencillos ya que posee lados rectos y no curvos.

Figura 3.4 Base Fija del Robot Delta acoplamiento



Vista superior de la base fija con sus respectivas medidas. Elaborado por: Esteban Martínez

Figura 3.5 Base Fija del Robot Delta implementada



Base fija final realizada en acrílico cortada con láser. Elaborado por: Esteban Martínez

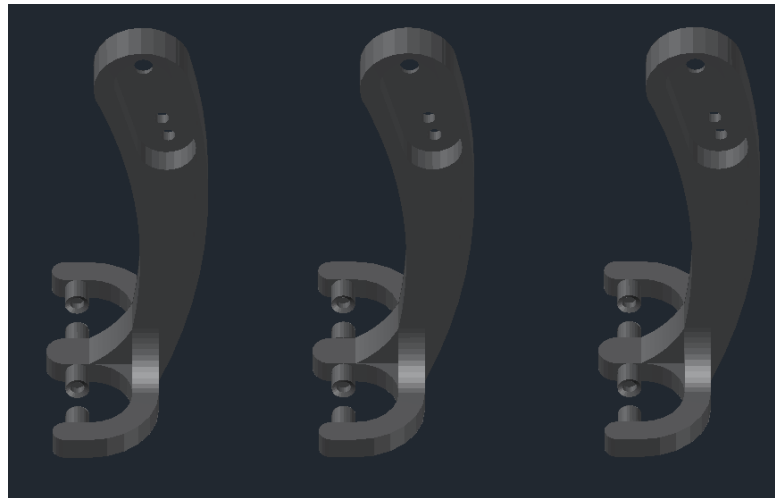
3.2.3 Aletas

Las aletas corresponden a la parte mecánica que se unen a los servos formando una sola pieza, las cuales se realizaron por medio del método de impresión 3D, las cuales requieren este método de impresión debido a su forma, reduciendo costos, brindando flexibilidad y dureza, ideales para reemplazar partes mecánicas de acero utilizadas en otros modelos de robot Delta.

3.2.3.1 Diseño de las Aletas

El diseño de las aletas del robot se las realizó en el entorno AutoCad 2015, una vez tomadas las medidas, aplicando ingeniería inversa usada en el robot comercial para el diseño de la misma. Estas piezas junto con los brazos del robot son los de mayor precisión en su corte ya que si no se tiene un acabado correcto de las mismas no podrán encajar entre sí y peor aún no tendrán un movimiento coordinado que permita la correcta maniobra de la plataforma Delta.

Figura 3.6 Diseño de aletas



Vista isométrica de las aletas diseñadas en Autocad 2015. Elaborado por: Esteban Martínez

3.2.3.2 Construcción de Aletas

Una vez concluido el diseño de las aletas se procedió a examinar el material necesario para su implementación se determinó que el método de impresión 3D es el más adecuado debido a la complejidad que presentan las piezas y sobre todo como estas deben tener armonía entre las demás estructuras para su correcto funcionamiento, las aletas se deben acoplarse a los servos motores mediante las partes que incluye el mismo actuador estas son sus roscas finales embonando con las aletas como se aprecia en la Figura 3.7 (a) y después mediante rulimanes las aletas se unirán a los brazos del robot formando una sola estructura como se aprecia en la Figura 3.8 (a).

Figura 3.7 Aletas implementadas mediante impresión 3D



(a) Instalación de tornillos de sujeción acoplados al servo motor

(b) Aleta impresa en 3D.

Elaborado por: Esteban Martínez

Figura 3.8 Acople de ruliman en aleta Impresa en 3D



(a)

(b)

(a) Instalación de ruliman en aleta impresa en 3D (b) ruliman acoplado en aleta impresa en 3D.

Elaborado por: Esteban Martínez

Como se puede apreciar en la Figura 3.7 (a). los rulimanes acoplan con la aleta mediante presión es por eso la necesidad de un óptimo acabado, para finalizar en la Figura 3.7 (b). el acople está sujeto a la aleta esperando por el brazo para así formar una junta universal.

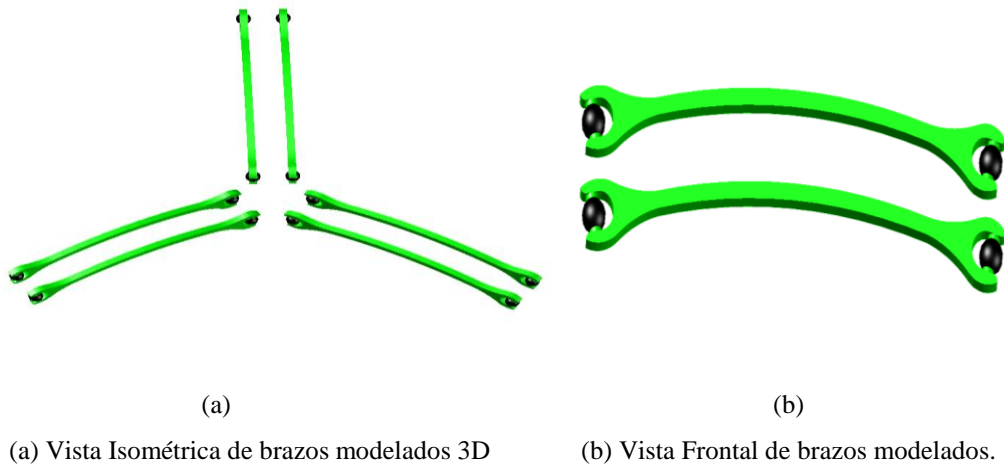
3.2.4 Brazos

Los Brazos conforman la parte mecánica que se unen con las aletas mediante las juntas universales, en nuestro caso serían los rulimanes acoplados a las mismas, cabe mencionar que junto con el servo motor acoplado a la base fija, la aleta se une en un extremo a los servos motores mientras que por el otro extremo mediante rulimanes a los brazos del robot estos conforman una cadena cinemática, en total tendríamos 3 cadenas cinemáticas las cuales se conectarán a la base móvil.

3.2.4.1 Diseño de los Brazos

El método que se ajustó a las características de un material duradero, flexible y liviano fue el de impresión 3D, también para agilizar la construcción de los mismos, una vez obtenidos los diseños mediante ingeniería inversa de los brazos se obtuvo el siguiente resultado como se puede apreciar en la Figura 3.9 (a). en la plataforma de AutoCad 2015.

Figura 3.9 Diseño de Brazos en AutoCad2015



Elaborado por: Esteban Martínez

3.2.4.2 Construcción de los Brazos

Los brazos junto con las aletas son una parte fundamental en el robot, de cómo estos deberán acoplarse armónicamente mediante las juntas universales, en nuestro caso los juntas universal vendrían a ser definidas como los rulimanes que actúan de manera similar, la cadena cinemática formada por aletas y brazos a través de rulimanes será ejecutada por cada servomotor ubicado en la plataforma fija, y esta cadena moverá en un plano x y o z al efector final, es por eso que se hará uso del método de impresión 3D mencionado anteriormente y usado en la mayoría de robot debido a su versatilidad, rapidez y resultado como se puede apreciar en la Figura 3.10.

Figura 3.10 Diseño de Brazos en AutoCad2015



Brazos Robóticos Implementados con el método de impresión 3D. Elaborado por: Esteban Martínez

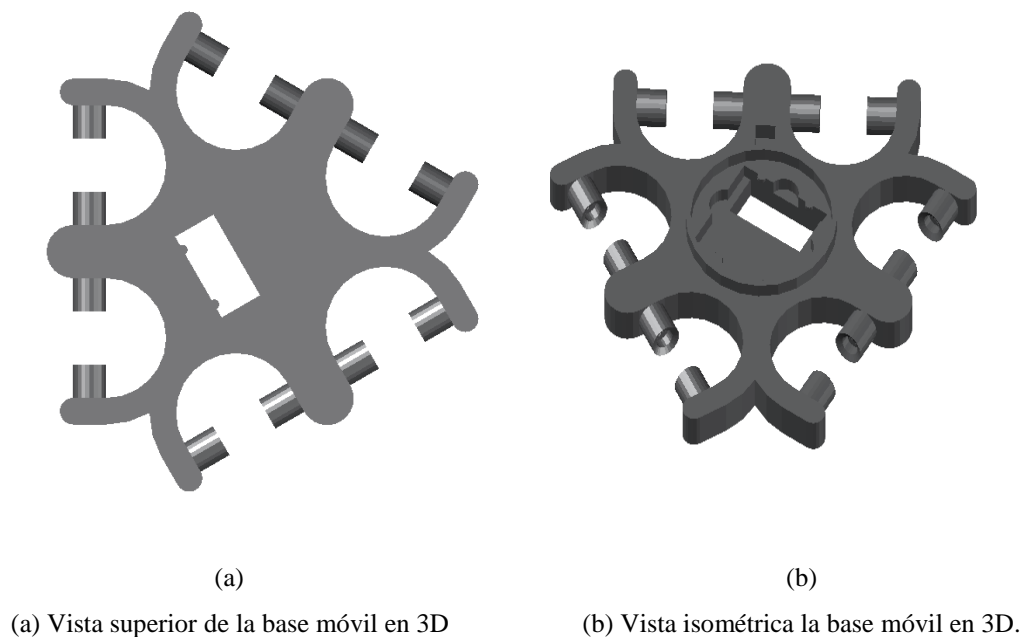
3.2.5 Base Móvil

Una de las piezas más importantes de la plataforma Delta es la base móvil en donde se ubicará el efector, en donde también se ubicará la pinza. También forma las uniones con los brazos del robot, las medidas y espacios están distribuidas según indique la morfología del robot Delta, el efector se situará en el extremo final de los brazos acoplados con la base móvil que nos permitirá interactuar con su entorno.

3.2.5.1 Diseño de la base Móvil

El software AutoCad nos permite realizar el diseño para la construcción de la base móvil después de haber obtenido las medidas mediante ingeniería inversa, como se aprecia en la Figura 3.11.

Figura 3.11 Diseño de Base móvil en AutoCad2015



Elaborado por: Esteban Martínez

3.2.5.2 Construcción de la Base Móvil

Como se aprecia en la Figura 3.12. el acabado de la impresión 3D es óptimo lo cual nos accede a acoplar todas las piezas para armar la plataforma Delta.

Figura 3.12 Base móvil impresa



Impresión en 3D de la base Móvil con acabados óptimos. Elaborado por: Esteban Martínez

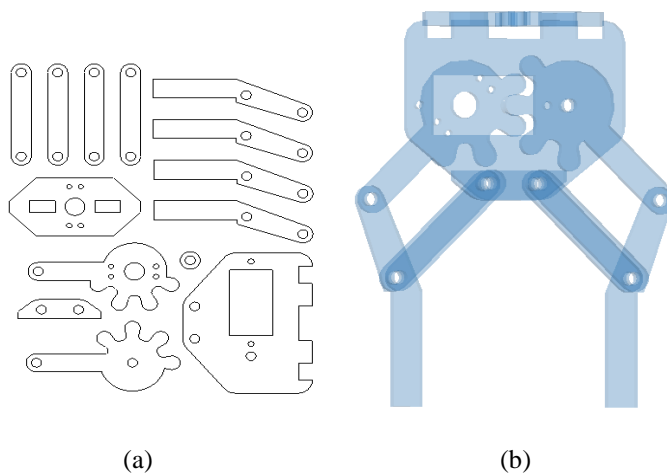
3.2.6 Gripper o Pinza

La pinza presente en el robot Delta replicará el cierre y apertura de mano capturado por el sensor Kinect, por lo que esta pinza será muy simple en su diseño, estará unida a la base móvil completando la estructura final del robot.

3.2.6.1 Diseño de la pinza o Gripper

De la misma manera que en los anteriores diseños el software AutoCAD nos brindara la facilidad de realizar el modelado de la pieza necesitada para posteriormente ser construida Figura 3.13.

Figura 3.13 Pinza Diseñada en el entorno de AutoCAD



(a) Gripper en etapa de modelado 2D

(b) Gripper modela mediante AutoCad 3D.

Elaborado por: Esteban Martínez

3.2.6.2 Construcción de la pinza o Gripper

El modelo diseñado en el entorno de AutoCAD será impreso en 3D cada pieza individualmente para posteriormente ser armada Figura 3.14. y posteriormente unida al resto del robot.

Figura 3.14 Pinza Ensamblada



Pinza implementada mediante el método de impresión 3D. Elaborado por: Esteban Martínez

3.3 Ensamblaje Total

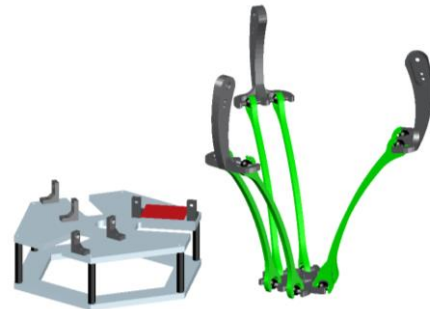
Como se aprecia en la Figura 3.15. las piezas modeladas en AutoCAD, han sido fusionadas, dando como resultado la estructura del robot parcial, en la Figura 3.15(a). se puede apreciar que la base fija del robot a un no ha sido fusionada, por otro lado, en la Figura 3.15(b)., se denota la parte de la base inmóvil.

Figura 3.15 Robot Delta Parcial Diseñado en el entorno de AutoCAD



(a)

(a) Estructura Parcial del Robot Delta



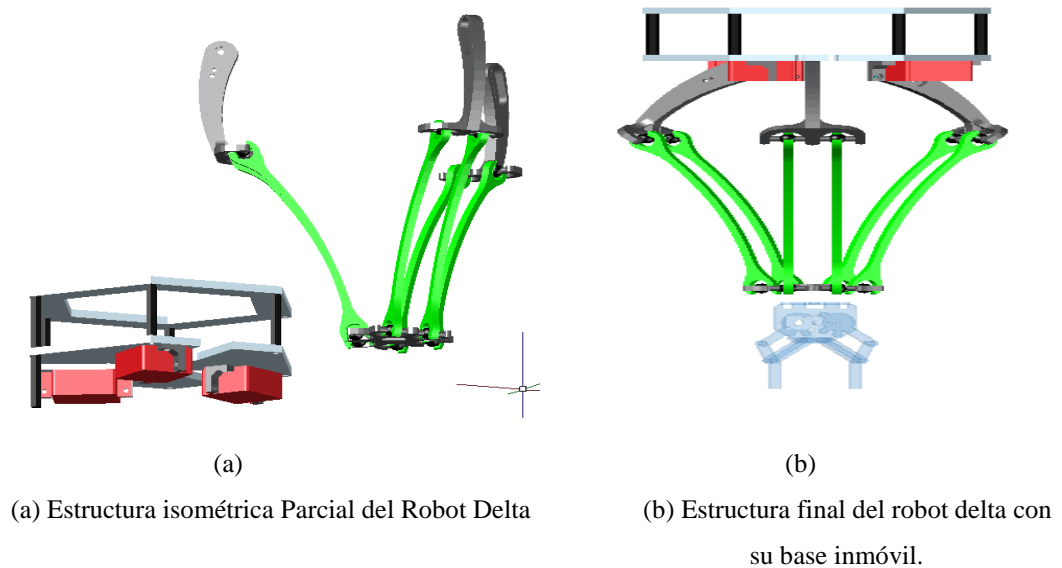
(b)

(b) Estructura parcial del robot delta con su base inmóvil.

Elaborado por: Esteban Martínez

En la Figura 3.16(b), el robot delta modelado en el entorno de Autocad, se encuentra finalizado, fusionado con su parte inmóvil

Figura 3.16 Robot Final Diseñado en el entorno de AutoCAD



Elaborado por: Esteban Martínez

Una vez realizado el bosquejo del ensamblaje del robot se procederá a realizarlo físicamente precautelando la cohesión de las piezas impresas mediante 3D Figura 3.17.

Figura 3.17 Robot Final Delta implementado



Estructura del Robot Delta Correctamente implementado. Elaborado por: Esteban Martínez

3.4 Desarrollo del Hardware Electrónico

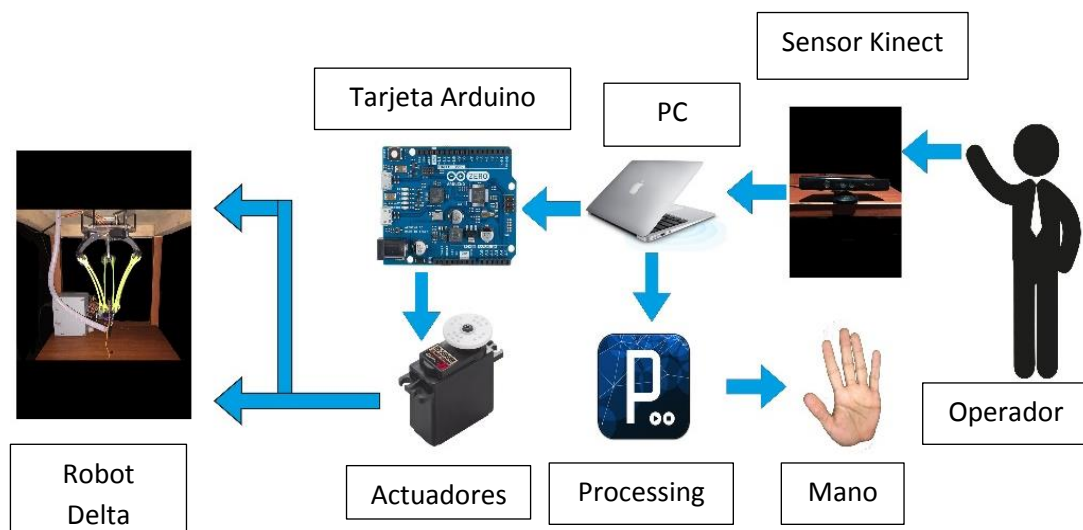
En esta sección se detallará el hardware electrónico necesario para cumplir con el objetivo de desarrollo del proyecto, en donde se especifica el diagrama general de

control, la tarjeta de actuación que interactúa con el software de Processing y el Robot Delta y el diagrama de conexión de los dispositivos.

3.4.1 Esquema General del Sistema

En la Figura 3.18 se presenta el esquema general que genera una propuesta de manipulación de la plataforma Delta mediante la detección de la postura y coordenadas de la mano de un operario, como sistema de adquisición de datos o de imágenes en este caso se usa el sensor Kinect fabricado por Microsoft, El algoritmo para la interacción de los datos entregados por el sensor se los realiza mediante el software abierto, específicamente con el programa Processing, este último enfocado para sistemas de diseño gráfico usado mayoritariamente en el desarrollo de proyectos con el sensor Kinect y para finalizar el medio de actuación en el que se presenta la versatilidad de funcionamiento de la tarjeta Arduino con cualquiera de sus variantes en tarjetas de control.

Figura 3.18 Esquema General de control del sistema del Robot Delta



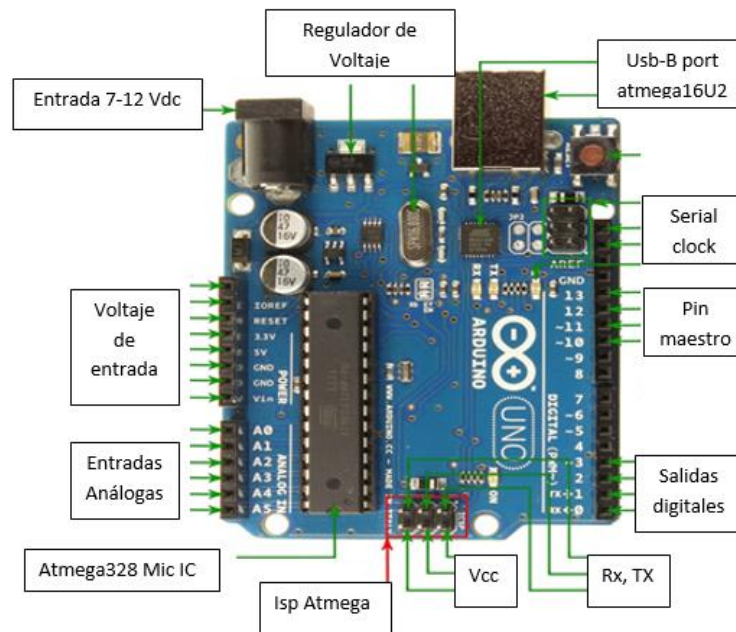
Esquema General de Control. Elaborado por: Esteban Martínez

3.4.2 Tarjeta de Actuación Maestro Arduino.

Para el caso de elaboración del Robot Delta se tomó la iniciativa de usar la tarjeta Arduino Uno como tarjeta de actuación entre el robot Delta y el entorno de Processing, debido a su bajo costo a comparación de otras plataformas que integran

microcontroladores, su software es multiplataforma es decir que funciona tanto en Windows como en Mac, su entorno de programación es muy simple y directo, fácil para principiantes y flexible para usuarios avanzados y su hardware cuenta con entradas y salidas de tipo análogas y digitales con una resolución de 10 bits un valor razonable para completar cualquier tipo de proyecto ya sea del tipo casero o del tipo industrial, todas las especificaciones técnicas de la tarjeta Arduino Uno se detallan en el Anexo C. La tarjeta de actuación Arduino Uno será la encargada de enviar las órdenes hacia los servomotores desde Processing una vez que este haya obtenido los parámetros de coordenadas desde nuestro analizador de datos, que en este caso sería el sensor Kinect Figura 3.19.

Figura 3.19 Diagrama electrónico Tarjeta Arduino Uno



Esquema Electrónico de la tarjeta Arduino Uno con sus respectivas salidas usadas en la plataforma Delta. Elaborado por: Esteban Martínez

En la Tabla 3.1 se muestra los pines de conexión que tiene el micro controlador de la tarjeta Arduino Uno con los componentes electrónicos que conforman el robot Delta.

Tabla 3.1 Pines de conexiones de la tarjeta Arduino Uno

Servo Pin	Conexión	Arduino Pin	Conexión
3	Servo Gripper 1	6	Salida PWM
3	Aux 1	5	Salida PWM
3	Servo Delta 3	9	Salida PWM
3	Servo Delta 2	10	Salida PWM
3	Servo Delta 1	11	Salida PWM

Tabla de conexionado referente a la tarjeta Arduino Uno

Elaborado por: Esteban Martínez

Para la parte de control de los servomotores 3 pertenecen al robot delta, 1 por cada articulación, mientras que los otros restantes forma parte del gripper además de un pin auxiliar, todos estos interactuando directamente con la tarjeta Arduino uno como se muestra en la Figura 3.19.

El microcontrolador en la placa Arduino se programa mediante lenguaje de programación C++ (este lenguaje también está basado en wiring), además en el entorno del desarrollo de la placa Arduino (basado en lenguaje similar al que posee Processing), cuenta con la facilidad que logra compilar los proyectos sin la necesidad de usar un ordenador.

3.4.3 Tarjeta de Control Auxiliar de alimentación

Además de la tarjeta de control maestra Arduino se necesitó de una tarjeta de control auxiliar de alimentación, la cual brinda la potencia, corriente y voltaje necesario que permiten accionar a los servomotores de una manera óptima ya que la corriente que brinda la tarjeta de control Arduino no suministra la energía necesaria para su funcionamiento eficaz. Para disminuir los gastos se decidió usar una tarjeta de control de energía de diseño propio.

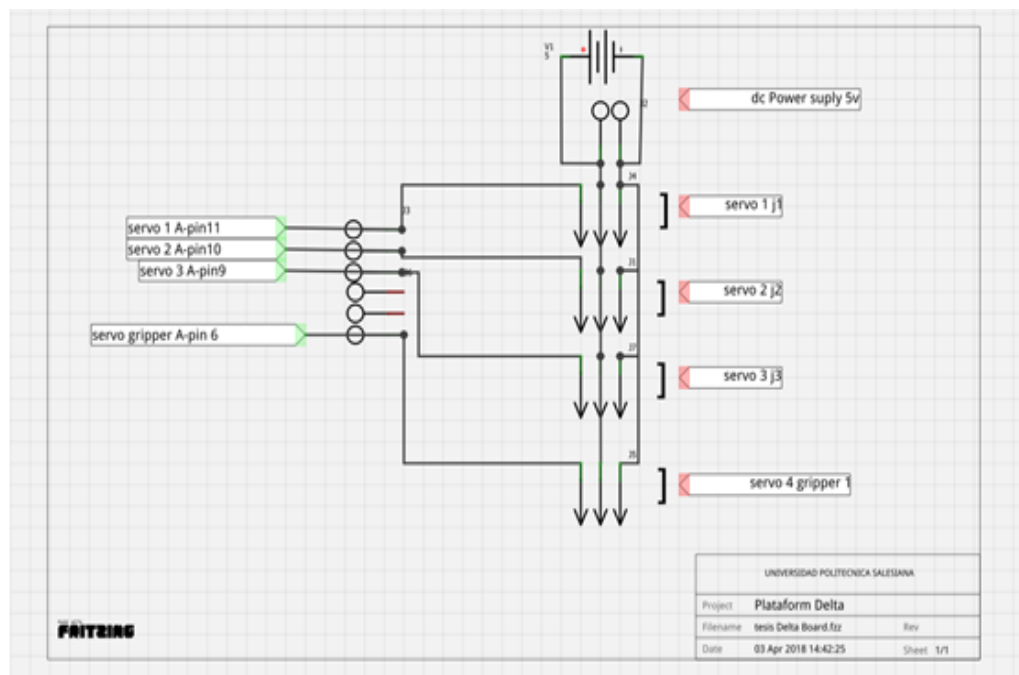
3.4.3.1 Diseño de la tarjeta de Control Auxiliar de alimentación

Para el diseño de la tarjeta de control de alimentación se utilizó el software de Fritzing, el cual por su fácil manejo y su amplio catálogo de dispositivos permite la fácil y rápida

elaboración de cualquier circuito electrónico. Para iniciar el diseño de la tarjeta es necesario dimensionar los sockets a utilizar tomando en cuenta el número de pines de los servos motores, cada servo motor cuenta con tres pines, de los cuales dos son alimentación +5 voltios, y uno pertenece a la señal PWM que controla al servo desde el Arduino.

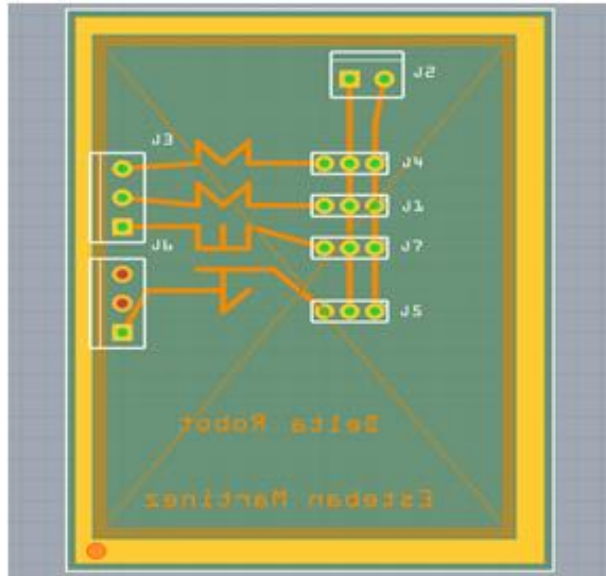
Después de elegir los sockets se realiza la simulación de pistas con una fuente de alimentación externa de 5V a 2 A conectado a los servos de la plataforma Delta y estos a su vez comandados por la tarjeta maestra Arduino Uno como se muestra en la Figura 3.20. Los servos motores presentes en la plataforma Delta, 3 de ellos en la base fija o inmóvil del robot son controlados mediante pulsos enviados desde la tarjeta Arduino Uno, mientras que el otro servo motor presente en la pinza, es controlado desde otro socket, pero de la misma forma con pulsos PWM.

Figura 3.20 Esquema electrónico Tarjeta de control de alimentación



Esquema Electrónico de conexiones hacia el robot Delta. Elaborado por: Esteban Martínez

Figura 3.21 PCB de la tarjeta de Control de alimentación



Tarjeta de control de alimentación PCB para el robot delta bottom Side realizado en el entorno Fritzing. Elaborado por: Esteban Martínez

Para el diseño del circuito PCB (Printer Circuit Board) se lo realizo con la herramienta de Fritzing, el cual permite realizar el diagrama esquemático de cada uno de los elementos utilizados y después realizar el ruteado de forma manual o automática con la opción de auto ruteado Figura 3.21.

Tabla 3.2 Pines de conexiones positivas de la tarjeta de control de Alimentación.

Servo Pin	Conexión	Tarjeta de alimentación	Conexión
2	Servo Gripper 1	+	Fuente +5v
2	Aux 1	+	Fuente +5v
2	Servo Delta 3	+	Fuente +5v
2	Servo Delta 2	+	Fuente +5v
2	Servo Delta 1	+	Fuente +5v

Tabla de conexión referente a la tarjeta de control de alimentación

Elaborado por: Esteban Martínez

Tabla 3.3 Pines de conexiones negativas de la tarjeta de control de Alimentación

Servo Pin	Conexión	Tarjeta de alimentación	Conexión
1	Servo Gripper 1	-	Fuente 0v
1	Aux 1	-	Fuente 0v
1	Servo Delta 3	-	Fuente 0v
1	Servo Delta 2	-	Fuente 0v
1	Servo Delta 1	-	Fuente 0v

Tabla de conexionado referente a la tarjeta de control de alimentación

Elaborado por: Esteban Martínez

3.4.3.2 Construcción de la tarjeta de control auxiliar de alimentación

Para el desarrollo del hardware electrónico se ejecutó el procedimiento de diseño de grabado y posteriormente erosionado, para lo cual el diseño se elaboró anteriormente en el entorno de Fritzing como se muestra en la Figura 3.21. Se procedió a imprimir el diseño en una hoja de papel termo transferible con la finalidad de estampar el dibujo en la baquelita, la cual fue previamente limpiada, después se procedió a quitar el cobre residual de la baquelita, esto se lo realiza con el proceso de revelado mediante ácido (cloruro férrico) en un recipiente además de contar con una brocha que ayuda a la remoción del cobre, hasta que solo quede el diseño realizado en Fritzing, Figura 3.22.

Figura 3.22 Tarjeta de Control auxiliar de alimentación terminada



Tarjeta de control de alimentación con sus elementos incorporados. Elaborado por: Esteban Martínez

3.5 Desarrollo del Software

En esta sección se hará mención a cada uno de los pasos ejecutados para el desarrollo y posterior implementación de los algoritmos para la lectura de procesamiento de imágenes obtenidos por el sensor Kinect interpretado por el entorno de Processing para después ser controlados por la tarjeta Arduino.

3.5.1 Diseño del Software

La plataforma Delta cuenta con un sistema de operación (imitación), el cual es ejecutado desde el entorno de Processing, este modo trata de replicar los movimientos de la mano del usuario que es identificada con el sensor Kinect moviendo los motores instalados en la plataforma fija del robot energizados con la tarjeta de control de alimentación auxiliar de una forma armónica a través de la tarjeta Arduino, entonces la visión artificial del sensor Kinect actúa capturando los movimientos, se podría decir que actúa como una tarjeta de adquisición de datos, después el entorno de Processing los interpreta y ejecuta una rutina de movimientos que son enviados a la tarjeta Arduino que como una tarjeta de actuación.

3.5.1.1 Tareas del Entorno de Processing

- Recibir los datos capturados provenientes del sensor Kinect
- Mostrar la información interpretada por el sensor Kinect
- Procesar la información para que pueda ser utilizada en la rutina de control.
- Establecer comunicación entre el entorno de programación y la tarjeta de control.
- Enviar los datos procesados a la Tarjeta de Control Arduino

3.5.1.2 Tareas de la tarjeta de Control Arduino

- Recibir los datos procesados desde el entorno de programación Processing
- Efectuar el proceso de locomoción según los datos recibidos desde el entorno de Processing.

3.5.1.3 Tareas de la tarjeta de control de alimentación Auxiliar

Proporcionar de los niveles óptimos de operación tanto en voltaje y corriente para que pueda ejecutar el correcto funcionamiento de locomoción.

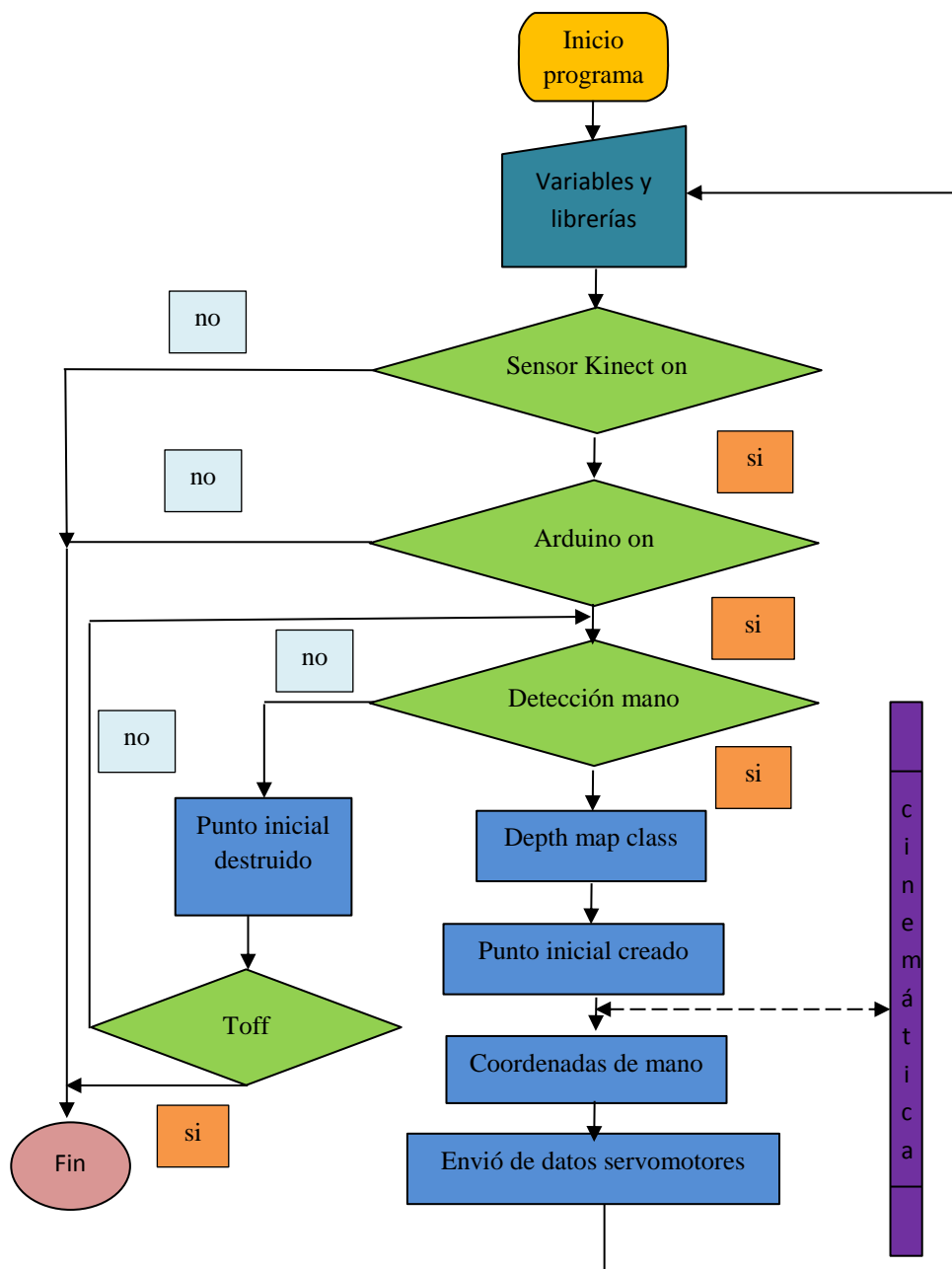
3.5.2 Implementación del Software

En esta sección se muestra el funcionamiento de los algoritmos implementados en el entorno de programación Processing y la Tarjeta de control Arduino.

3.5.2.1 Diagrama general de flujo del algoritmo del Robot Delta

Como se muestra en la Figura 3.23 se presenta el diagrama de flujo el cual indica el funcionamiento una vez se hayan iniciado las librerías, el sensor Kinect este encendido, y la tarjeta de Arduino este transmitiendo accionamiento a los actuadores. Este diagrama de flujo se desglosa en funciones específicas las cuales serán explicadas en secciones posteriores de este capítulo.

Figura 3.23 Estructura General del software para el entorno Processing



Representación del algoritmo de Control del Entorno de Processing. Elaborado por: Esteban Martínez

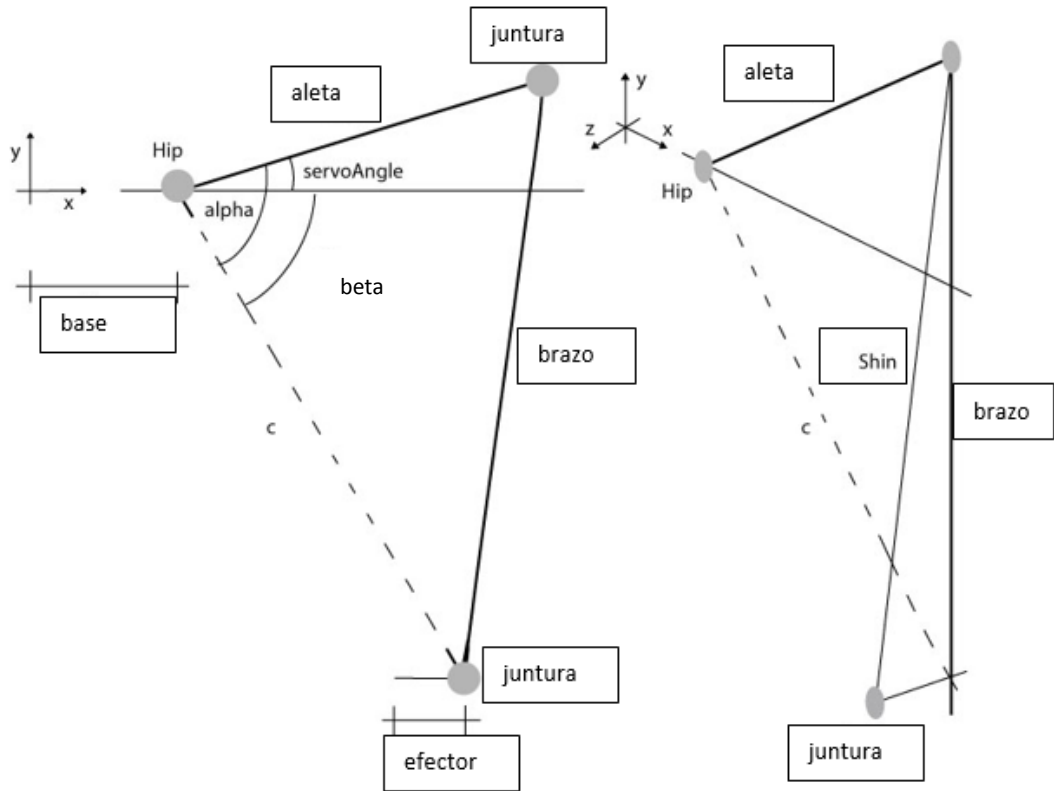
3.5.2.2 Análisis Cinemático Inverso a nivel de Software

Inicialmente se debe aclarar un concepto importante si se trabaja con robots, muy comúnmente se necesita usar la cinemática inversa y existen algunos problemas que se necesitan resolver antes de analizar sus movimientos. Hablando de manera puntual el problema que desarrollaremos a continuación viene desencadenado con una pregunta: ¿Cuáles son los estados requeridos de los dispositivos actuadores si quiero lograr una posición específica?

Entonces si se desea mover la base móvil a puntos requeridos en un lugar determinado que pueda imitar los movimientos de la articulación humana, si bien es cierto se dispone de una estructura simple la cual está moviéndose a través de los servomotores colocado en la base inmóvil de nuestra plataforma hay que conocer como estos tendrán que interactuar entre si moviendo coordenadas en los espacios de x, y, z y al final definirán la posición deseada de la base móvil, la única manera es haciendo un análisis de la cadena cinemática y de la geometría de nuestro robot y sobre todo las limitaciones que esta genera sobre la cadena cinemática.

Se buscará la cinemática inversa de un brazo del robot para así encontrar el movimiento del servo que alcance una posición apropiada determinada en el espacio, pero el análisis debe ser orientado a una estructura tridimensional que al final nos indicará la ecuación correcta a utilizar. (Flacco Fabrizio, 2017)

Figura 3.24 Análisis geométrico



Análisis geométrico del Robot Delta. Elaborado por: Esteban Martínez

Observando la Figura 3.24 se define que el problema estructural tridimensional ha quedado reducido a dos dimensiones debido a que el problema queda ideado a su proyección sobre el plano de trabajo del servo de rotación obteniendo la respuesta mediante la geometría plana.

$$\text{float } c = \text{dist}(\text{postemp.varx} + \text{effectorsize}, \text{postemp.vary}, \text{basesize}, 0); \quad \text{Ec. (3.1)}$$

$$\text{float } \alpha = \text{acos}((-a2 + \text{thigh} * \text{thigh} + c * c) / (2 * \text{thigh} * c)) \quad \text{Ec. (3.2)}$$

$$\text{float } \beta = -\text{atan2}(\text{postem.y}, \text{postemp.x}); \quad \text{Ec. (3.3)}$$

Para conocer el valor de α se necesita conocer el valor de $a2$, la ecuación que define a $a2$ se presenta a continuación, entonces $a2$ quedaría expresada de la siguiente manera:

$$\text{float } a2 = \text{shin} * \text{shin} - \text{poztemp.z} * \text{poztemp.z} \quad \text{Ec. (3.4)}$$

Una vez resuelto $a2$ nos quedaría remplazar el resultado en la ecuación de α para conocer su valor, si se conoce a α y β , podremos encontrar al ángulo del servo al cual llamaremos θ , en donde finalmente;

$$\text{servoAngle } \theta = \text{alpha} - \text{beta}; \quad \text{Ec. (3.5)}$$

3.5.2.3 Diagrama de flujo para el control del Robot

La Figura 3.25 como se muestra a continuación indica el funcionamiento del algoritmo del control del robot Delta una vez este entra en el proceso posterior de adquisición de datos, durante la ejecución de la rutina y de envío de datos seriales.

Figura 3.25 Flujograma utilizado para el control del robot

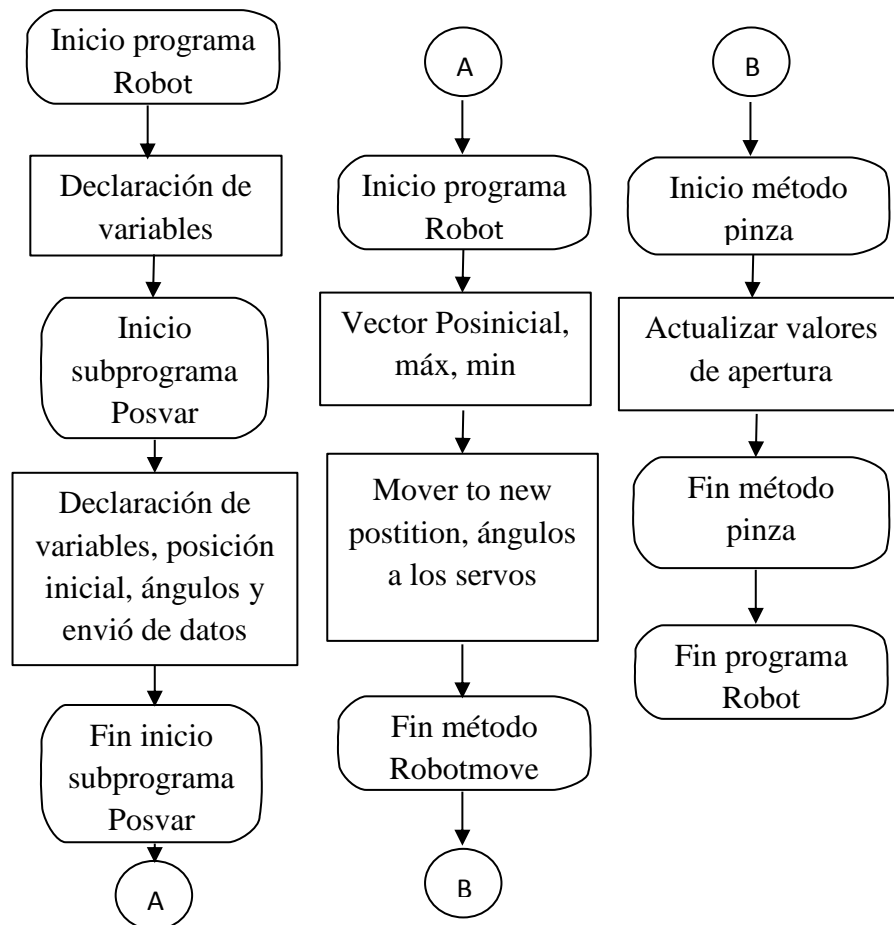


Diagrama de flujo control de robot en el entorno de programación. Elaborado por: Esteban Martínez

3.5.2.4 Diagrama de flujo para el control de los brazos o Piernas

En el diagrama de flujo anterior se ha implementado las rutinas principales para poder dirigir el robot, a continuación, se implementará la rutina de la cinemática inversa del robot en donde se define al ángulo θ de los servos que define la posición deseada.

Figura 3.26 Flujograma empleado en el control de control de los brazos

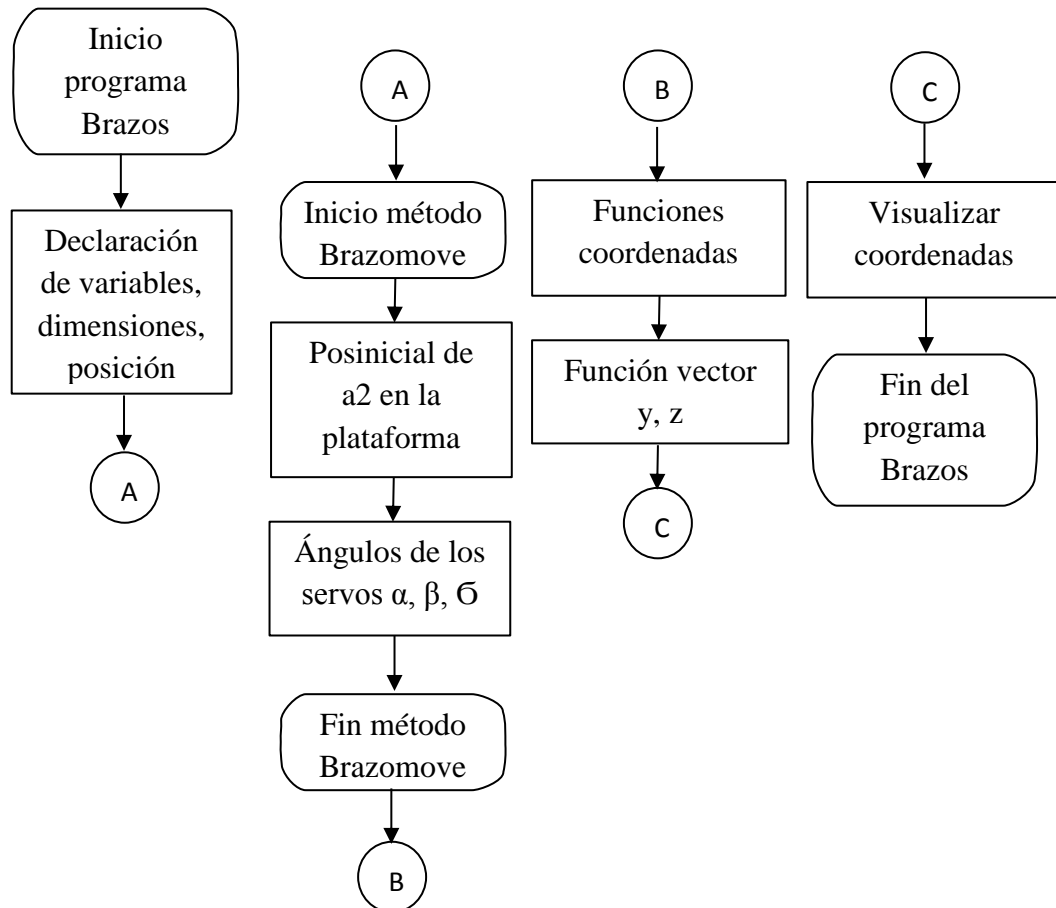


Diagrama de flujo control de brazo en el entorno de programación. Elaborado por: Esteban Martínez

3.5.2.5 Diagrama de flujo para Kinect y Delta

Con las dos rutinas implementadas se pretende desarrollar una tercera que aplique las características de imitación o seguimiento de la extremidad superior en este caso mediante la librería de Kinect OpenNite para desplazar el robot Delta. Dentro de esta tercera rutina estará la programación de comunicación con la tarjeta de control Arduino que sirve como puente entre el Delta físico y su interacción con el miembro superior.

Figura 3.27 Flujograma utilizado para el control de los brazos

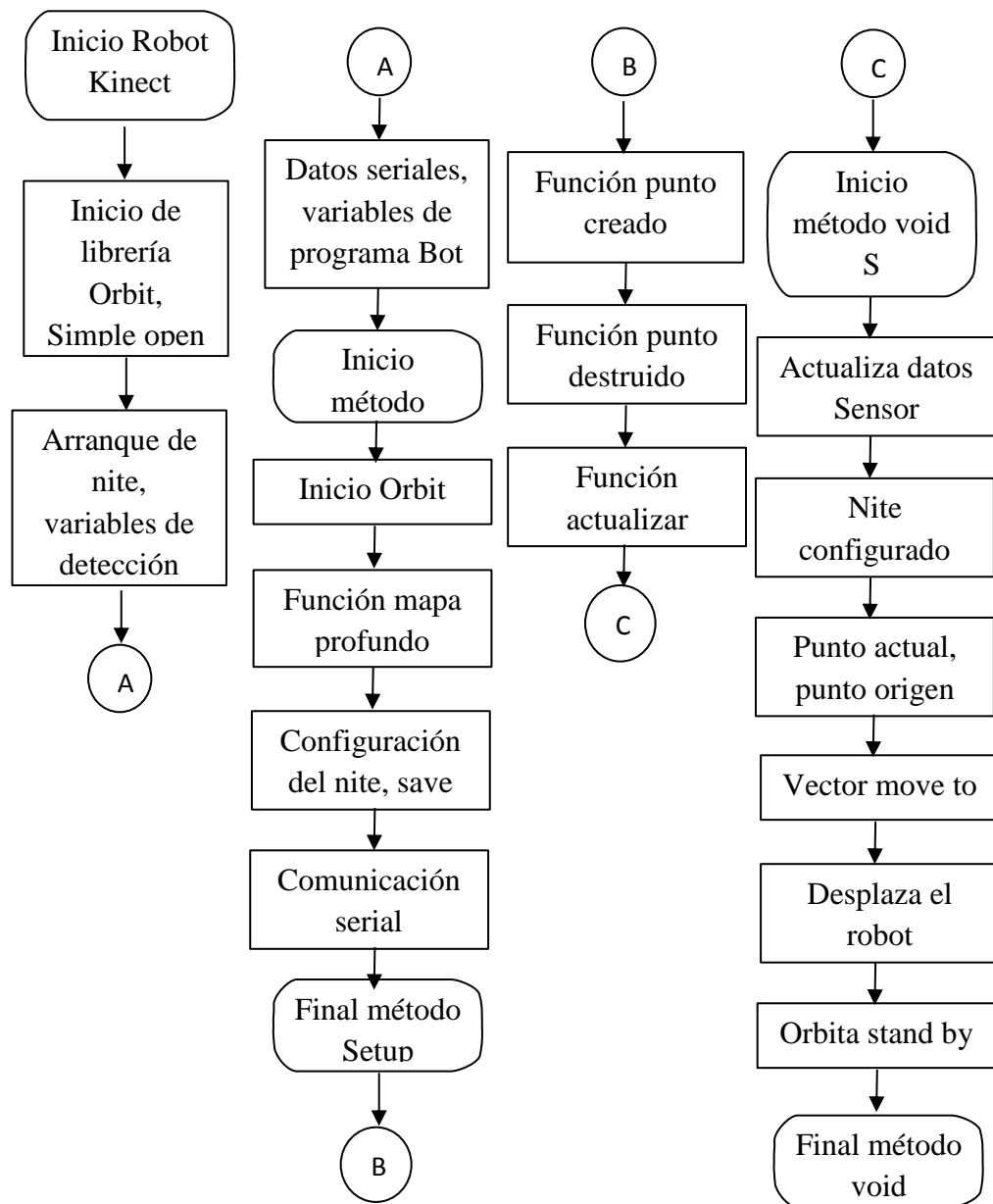


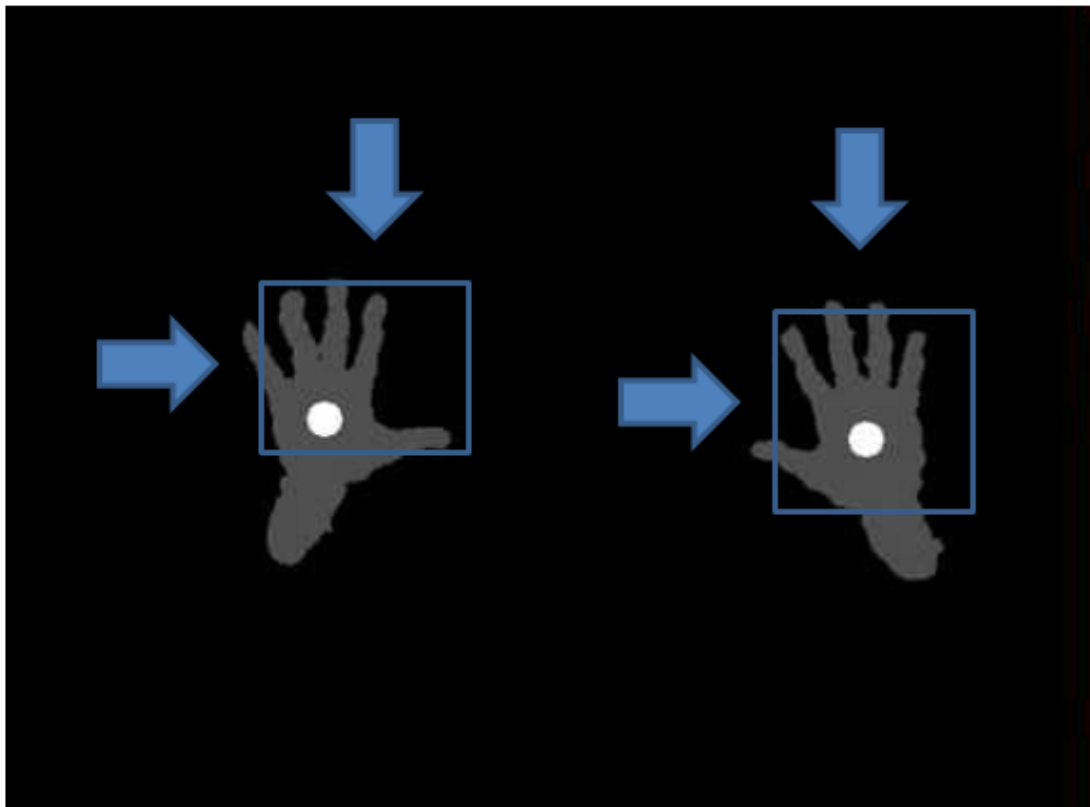
Diagrama de flujo control de brazo en el entorno de programación. Elaborado por: Esteban Martínez

3.5.2.6 Apertura y cerrado de Gripper o pinza

La rutina de adquisición procesamiento y actuación, se ha desarrollado, como parte final la implementación de la rutina para el control de la pinza situada en la base móvil, como se ha optado por la adquisición del sensor Kinect v1, este en su software no dispone del reconocimiento de los dedos de la mano como una pinza, sin embargo eso no limita a que se pueda realizar instrucciones de censado mediante la librería de gestos (gesture), como se opta por esta opción la rutina se volverá un poco más extensa, el

primer paso a realizar es que la mano del usuario final podría definirse como un conjunto de puntos, y este sea procesado como una cloudpoints, esta se mediría cuando la mano se encuentra cerrada y de la misma manera cuando esta se encuentre abierta, como se aprecia en la Figura 3.28 si se ha identificado correctamente la distancia del cloudpoints al abrir será más grande y al cerrar la mano sucederá lo contrario.

Figura 3.28 Nube de puntos en Nite(cloudPoints)



Nube de puntos visualizada en el entorno de Processing. Elaborado por: Esteban Martínez

Para realizar la apertura y cerrado de la pinza se necesitan conocer algunas variables, posteriormente calcular sus distancias, se implementó el siguiente algoritmo representado en este diagrama de flujo.

3.5.2.7 Diagrama de flujo rutina de Gripper

Figura 3.29 Diagrama de flujo representado a la rutina de Gripper

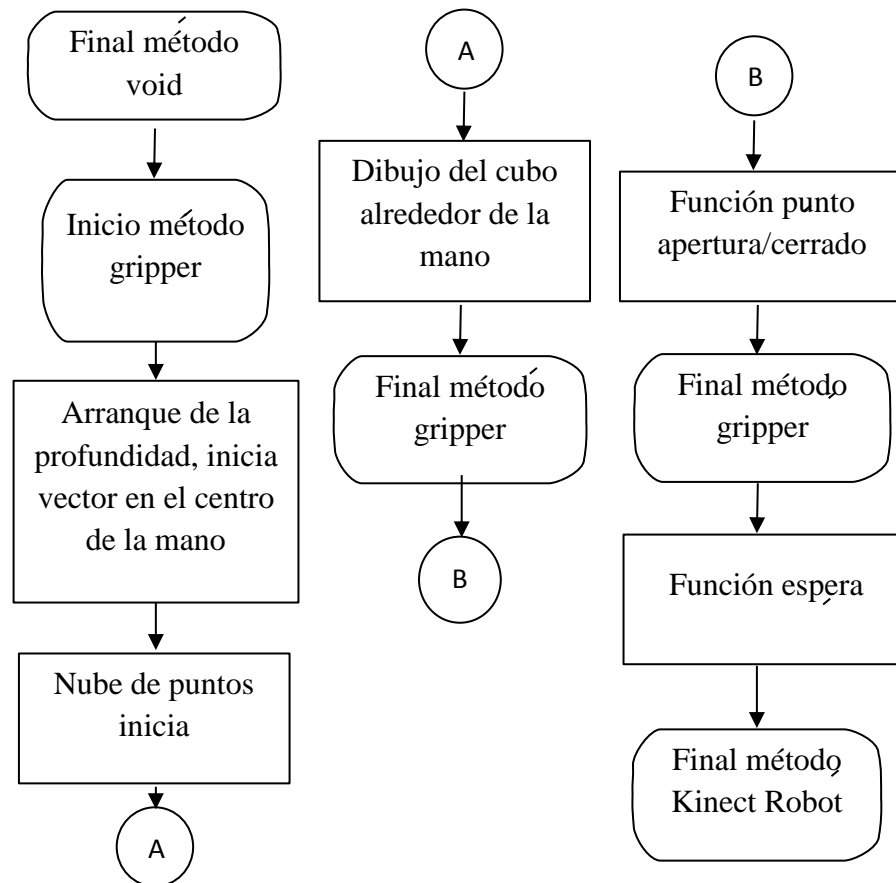


Diagrama de flujo de Gripper en el entorno de programación. Elaborado por: Esteban Martínez

3.5.2.8 Tarjeta de Control Arduino

La rutina implementada en la tarjeta Arduino posee una función simple la cual es leer los datos en serie enviados desde el entorno de Processing, el Arduino entonces procesa estos datos y los escala, posteriormente envía pulsos únicos a cada uno de los servomotores para que se puedan mover acorde al miembro superior.

Figura 3.30 Diagrama de flujo representado a la rutina de actuación en la tarjeta

Arduino

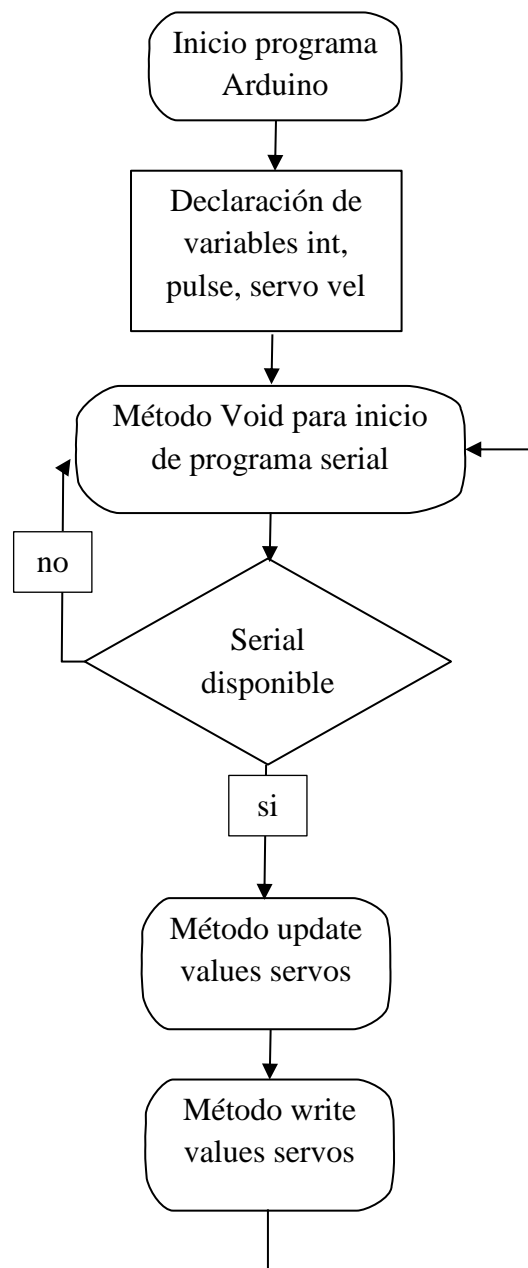


Diagrama de flujo para la tarjeta de actuación Arduino. Elaborado por: Esteban Martínez

CAPÍTULO 4

ANÁLISIS DE RESULTADOS

El siguiente capítulo describe cada una de las pruebas realizadas durante y al final de la construcción del robot además de los análisis que prosiguen a ellas, comprobando el funcionamiento aplicativo de la plataforma.

4.1 Pruebas de Hardware Mecánico

Al comienzo del proyecto se implementó ingeniería inversa del robot referencial Delta Army, del mismo se realizó el diseño en el entorno de AutoCad conforme al método de investigación y luego al de diseño, las juntas, los brazos, los servos motores, los elementos mecánicos y electrónicos conseguidos los cuales se ensamblan en la estructura, como se aprecia en la Figura 4.1.

Figura 4.1 Plataforma Delta



Prototipo de la Plataforma Delta. Elaborado por: Esteban Martínez

4.2 Pruebas de Tarjeta de Control

Para realizar la prueba de comunicación entre el software de Processing y la tarjeta de control Arduino, se la dividió en las siguientes partes:

- Comunicación Sensor Kinect - Processing
- Comunicación Processing - tarjeta de actuación Arduino

4.2.1 Comunicación Sensor Kinect-Processing

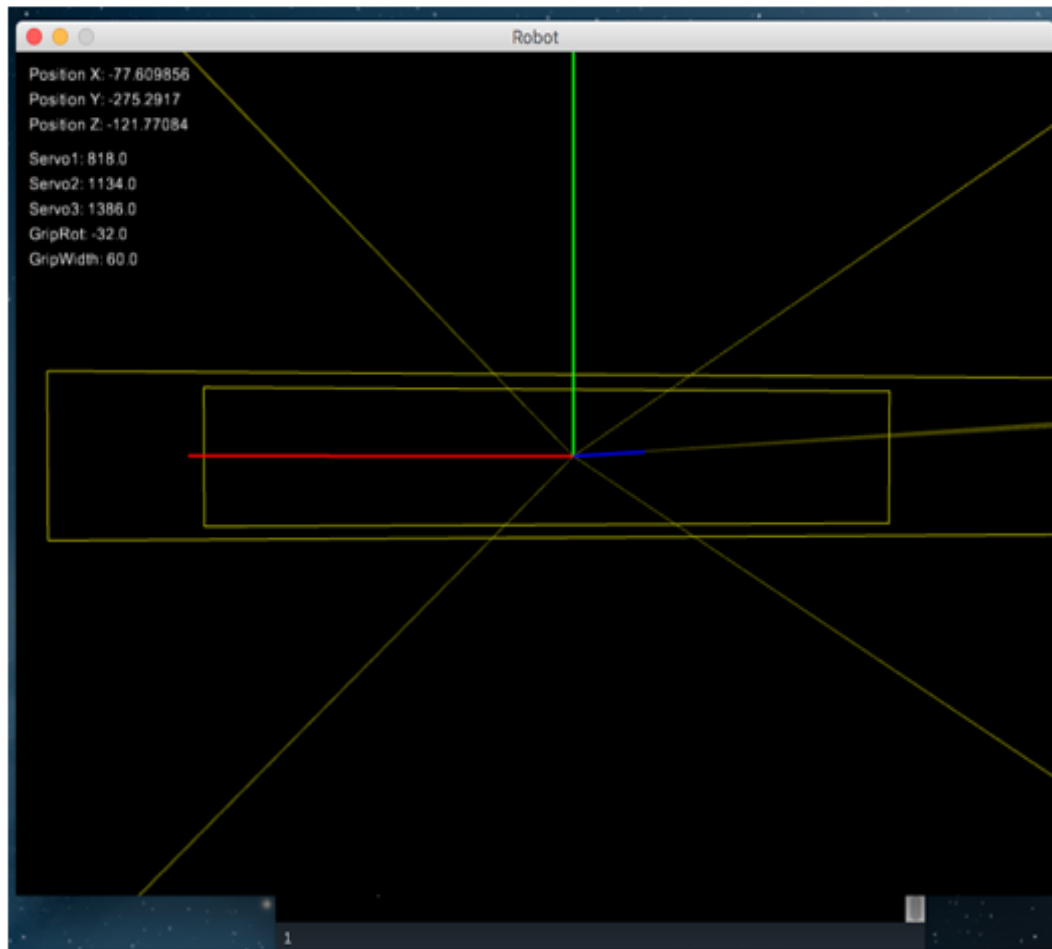
En esta Prueba se verifica el funcionamiento de la comunicación de recepción de datos capturados por el sensor de Microsoft Xbox Kinect V1.41 hacia el entorno de programación Processing.

Como se mencionó en el capítulo 3 en el esquema general, el sensor Kinect capturará los datos mediante visión artificial estos serán leídos por el entorno de Processing para que puedan ser usados en la rutina de control implementada en el software.

En esta prueba se debe evidenciar que el entorno de Processing, en su interfaz de presentación muestre los datos capturados después de que se haya procedido a compilar y el entorno este en modo runtime y que al movimiento del miembro superior, la mano en específico, los datos muestreados por el sensor Kinect puedan ser presentados de tal forma que puedan ser utilizados en la rutina de procesamiento y posteriormente en la rutina de control, tal como se muestra en la Figura 4.2.

Para esto es necesario que el sensor Kinect utilice las librerías adecuadas en el momento de la integración con el entorno de Kinect, en el Anexo B se da a conocer la librería utilizada para que se pueda establecer un medio de comunicación entre el software y el dispositivo, para el envío de datos capturados es necesario que el sensor Kinect sea el modelo de Xbox 360 V1.41, ya que si se optaría por el modelo del sensor Kinect V2 las librerías no estarían en sinergia con el entorno haciendo que se deba cambiar desde el hardware de comunicación, hasta las librerías importadas para el software.

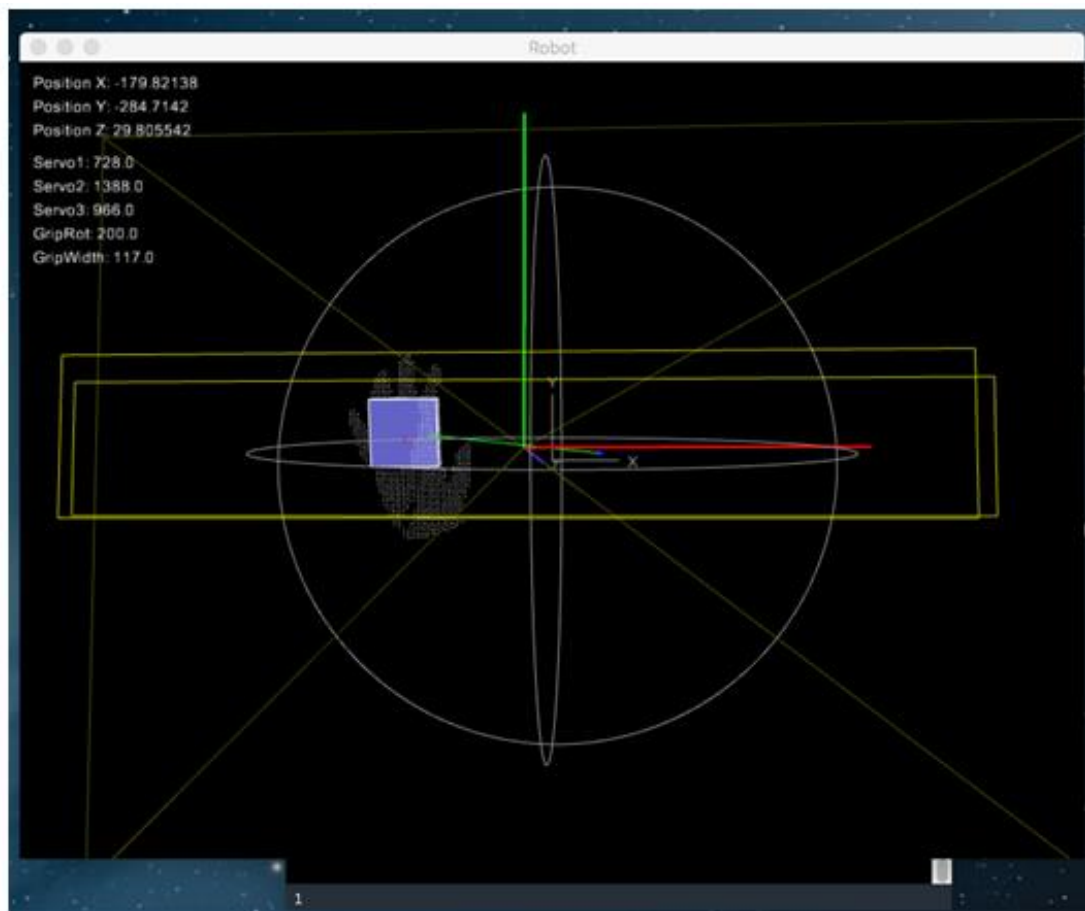
Figura 4.2 Interfaz de presentación de datos Capturados por el sensor Kinect



Datos presentados por el sensor Kinect V1.41 utilizados en la rutina de procesamiento. Elaborado por:
Esteban Martínez

De la misma manera en la interfaz se debe mostrar también los datos de la captura de la apertura y cierre de la mano realizados con la librería Gesture, en ella se dibuja un cuadrado alrededor de la mano con su máxima área cuando esta esté abierta, y con un área mínima cuando esta se encuentre cerrada con la que haremos actuar al gripper o pinza como se muestra en la Figura 4.3.

Figura 4.3 Interfaz de presentación de datos Capturados por el sensor Kinect para la pinza o gripper

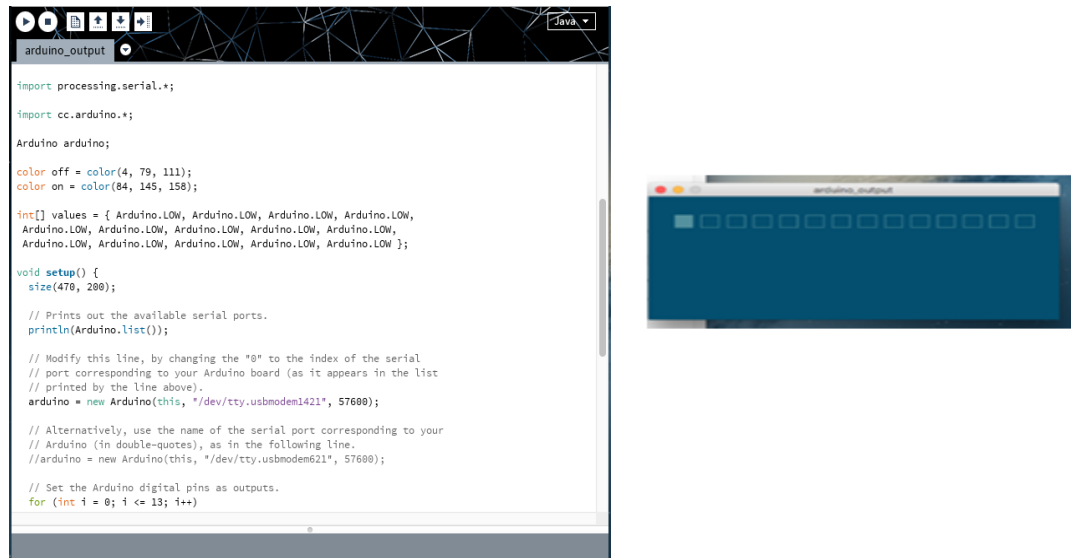


Datos presentados por el sensor Kinect V1.41 utilizados en la rutina de procesamiento para la pinza o gripper Elaborado por: Esteban Martínez

4.2.2 Comunicación Processing – Tarjeta de Actuación Arduino

Fue necesaria la incorporación de librerías de Arduino en el entorno de Processing para la comunicación entre ambos y de la misma manera en la tarjeta actuación Arduino se tuvo que agregar librerías de comunicación con el entorno, una vez incorporadas las librerías y se procede a ejecutar un ejemplo en el software con el que vamos a realizar el parpadeo del led principal de la tarjeta de actuación Arduino desde la interfaz de Processing, si es exitosa la prueba de comunicación se procede a ejecutar la rutina de control que envía datos a través de comunicación serial hacia la tarjeta de actuación Arduino como se muestra en la Figura 4.4

Figura 4.4 Datos enviados de manera serial hacia la tarjeta de Actuación

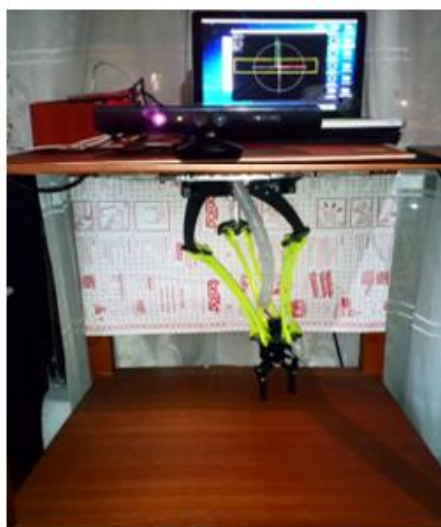


Datos enviados hacia la tarjeta de actuación Arduino Elaborado por: Esteban Martínez

4.3 Protocolos de Pruebas de Movimiento

Después de comprobar el funcionamiento de hardware de la comunicación entre el sensor Kinect - Processing - Arduino, se continua a realizar la prueba de movimiento de la plataforma Delta. Una vez entrado en el modo de ejecución desde el Processing el robot Delta busca la posición inicial, esperando a que se envíen datos capturados por el sensor como se muestra en Figura 4.5

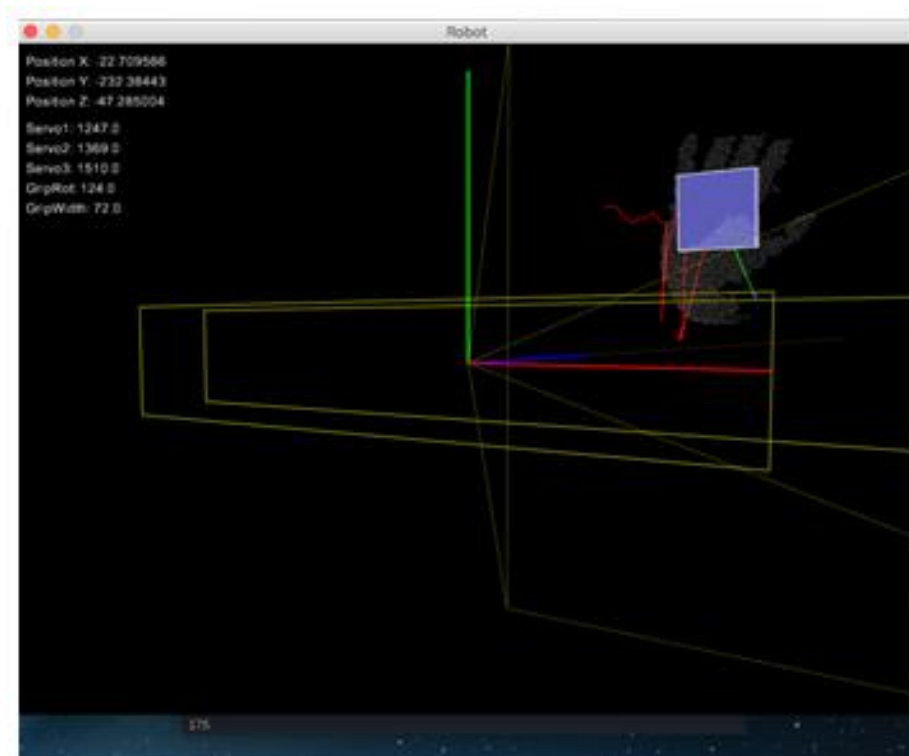
Figura 4.5 Ejecución de Rutina de movimientos para el Robot Delta



Robot Delta en posición inicial esperando movimiento. Elaborado por: Esteban Martínez

Una vez la mano entra en el área de captura del sensor se dibujará una mano en la interfaz del Processing la cual al movimiento mostrara una estela de color, indicando la captura del movimiento de la mano del usuario el proceso de movimiento del robot es seguir a la mano del usuario en el campo de captura, Figura 4.6.

Figura 4.6 Movimientos capturados por el Sensor



Movimiento del mano capturado por el sensor Kinect y reflejado en Processing.

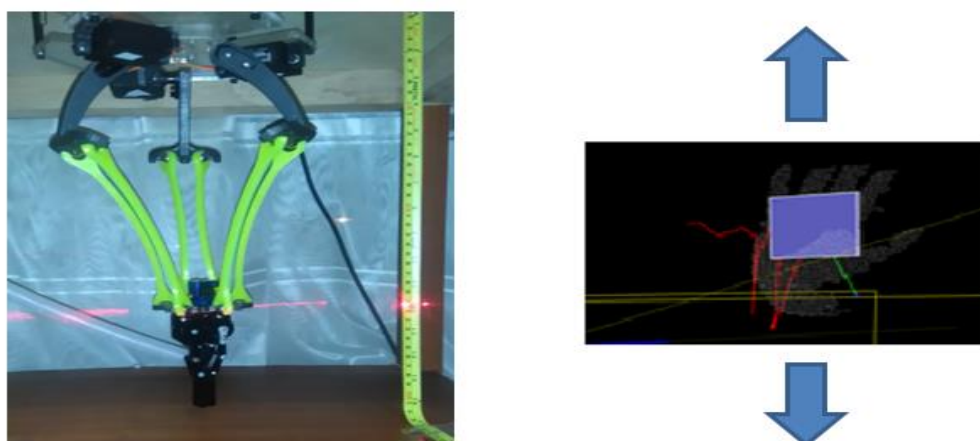
Elaborado por: Esteban Martínez

A continuación, se recopilará datos de análisis en base al movimiento de la plataforma Delta con respuesta a una trayectoria planificada básica. Se realizaron 6 pruebas básicas de las cuales las 3 primeras corresponden a movimientos de la mano que pretenderá determinar la precisión con movimientos simples siguiendo una trayectoria base.

4.3.1 Movimiento Vertical

Primera prueba básica que hace referencia al movimiento de la mano en el eje y, se procedió a subir y bajar la mano, para posteriormente tomar medidas, Figura 4.7.

Figura 4.7 Protocolo de pruebas de movimiento en el eje Y



Pruebas de movimiento vertical. Elaborado por: Esteban Martínez

Tabla 4.1 Valores obtenidos en la trayectoria trazada en el eje Y.

Trayectoria vertical dada por el patrón que sigue la mano vs Trayectoria vertical imitada por el robot Delta				
Movimiento en eje Y punto inicial de la trayectoria y punto final de la trayectoria				
Número de datos tomados	Movimiento de la mano		Movimiento del robot	
	Punto inicial	Punto final	Punto inicial	Punto final
1	1	30	1.1	15
2	1	30	1	15
3	1	30	1.1	15.05
4	1	30	1	15.05
5	1	30	1.12	15.05
6	1	30	1	15
7	1	30	1.13	15.05
8	1	30	1.12	15.05
9	1	30	1.1	15
10	1	30	1.08	15
11	1	30	1.15	15.05
12	1	30	1.12	15.1
13	1	30	1	15.05
14	1	30	1	15.05
15	1	30	1	15
16	1	30	1.13	15
17	1	30	1	15.05
18	1	30	1.12	15.5
19	1	30	1	15.5
20	1	30	1.1	15.5

Tabla de valores en el movimiento vertical. Elaborado por: Esteban Martínez

Tabla 4.2 Tabla de efectividad y errores en la trayectoria trazada en el eje Y.

Trayectoria vertical dada por el patrón que sigue la mano vs Trayectoria vertical imitada por el robot Delta			
Movimiento en eje Y punto inicial de la trayectoria y punto final de la trayectoria			
Tabla de resultados			
Error en Punto inicial	Error en Punto final	Error total %	Efectividad total %
0.1	0	0.67	99.33
0	0	0	100
0.1	0.05	1	99
0	0.05	0.33	99.67
0.12	0.05	1.13	98.87
0	0	0	100
0.13	0.05	1.2	98.8
0.12	0.05	1.13	98.87
0.1	0	0.67	99.33
0.08	0	0.53	99.47
0.15	0.05	1.33	98.67
0.12	0.1	1.47	98.53
0	0.05	0.33	99.67
0	0.05	0.33	99.67
0	0	0	100
0.13	0	0.87	99.13
0	0.05	0.33	99.67
0.12	0.5	4.13	95.87
0	0.5	3.33	96.67
0.1	0.5	4	96
		1.14	98.86
		Error promedio %	Efectividad promedio %

Tabla de valores de acierto y error en el movimiento vertical.

Elaborado por: Esteban Martínez

Para el cálculo del error absoluto presente en el punto inicial se utilizó la Ecuación 4.1, por otro lado, para el cálculo del error absoluto presente en el punto final se utilizó la Ecuación. 4.2, descritas a continuación.

$$\text{Error en punto inicial} = |(\text{valor real} - \text{valor medido})| \quad \text{Ec. (4.1)}$$

$$Error\ en\ punto\ final = |(valor\ real - valor\ medido)| \quad Ec.\ (4.2)$$

Una vez obtenidos los errores iniciales y finales se procede a utilizar la Ecuación 4.3 y la Ecuación 4.4, descritas a continuación para el cálculo del error total y la efectividad total respectivamente.

$$Error\ total = \frac{Error\ en\ punto\ inicial + Error\ en\ punto\ final}{valor\ real\ medido} \times 100\% \quad Ec.\ (4.3)$$

$$Efectividad\ total = Valor\ R - \left(\frac{Error\ Pi + Error\ Pf}{valor\ real\ medido} \right) \times 100\% \quad Ec.\ (4.4)$$

Para finalizar se procede a calcular el error promedio y la efectividad promedio utilizando la Ecuación 4.5 y la Ecuación 4.6.

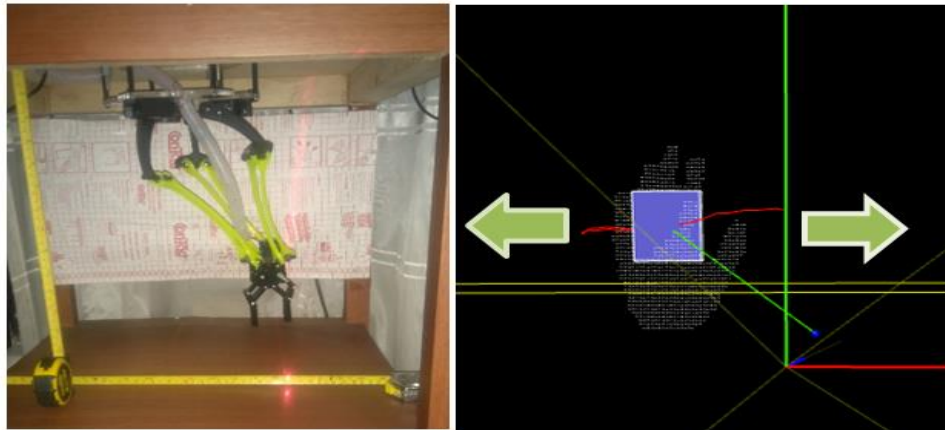
$$\% Error\ promedio = \sum \left(\frac{Error\ total}{Numero\ de\ pruebas\ realizadas} \right) \times 100\% \quad Ec.\ (4.5)$$

$$Efectividad\ Total = \sum \left(\frac{Efectividad\ total}{Numero\ de\ pruebas\ realizadas} \right) \times 100\% \quad Ec.\ (4.6)$$

4.3.2 Movimiento Horizontal

Segunda prueba básica que hace referencia al movimiento de la mano en el eje X, se procedió a mover la mano de izquierda a derecha, para posteriormente tomar medidas, para comprobar la prueba de precisión en un solo punto, Figura 4.8.

Figura 4.8 Protocolo de pruebas de movimiento en el eje X



Pruebas de movimiento Horizontal. Elaborado por: Esteban Martínez

Tabla 4.3 Valores obtenidos en la trayectoria trazada en el eje X.

Trayectoria vertical dada por el patrón que sigue la mano vs Trayectoria vertical imitada por el robot Delta				
Movimiento en eje X punto inicial de la trayectoria y punto final de la trayectoria				
Número de datos tomados	Movimiento de la mano		Movimiento del robot	
	Punto inicial	Punto final	Punto inicial	Punto final
1	18	48	18.2	34.8
2	18	48	18.3	34.5
3	18	48	18	34.71
4	18	48	18.2	34.6
5	18	48	18.1	34.8
6	18	48	18.5	34.6
7	18	48	18.6	34.9
8	18	48	18.1	34.5
9	18	48	18.3	34.6
10	18	48	18.5	34.7
11	18	48	18.5	34.7
12	18	48	18.3	34.9
13	18	48	18.3	34.6
14	18	48	18.3	34.7
15	18	48	18.5	34.9
16	18	48	18.2	34.5
17	18	48	18.1	34.8
18	18	48	18.4	34.6
19	18	48	18.2	34.6
20	18	48	18.5	34.7

Tabla de valores en el movimiento horizontal. Elaborado por: Esteban Martínez

Tabla 4.4 Tabla de efectividad y errores en la trayectoria trazada en el eje X.

Trayectoria vertical dada por el patrón que sigue la mano vs Trayectoria vertical imitada por el robot Delta			
Movimiento en eje X punto inicial de la trayectoria y punto final de la trayectoria			
Tabla de resultados			
Error en Punto inicial	Error en Punto final	Error total %	Efectividad total %
0.2	0.8	6.25	93.75
0.3	0.5	5	95
0	0.71	4.43	95.56
0.2	0.6	5	95
0.1	0.8	5.62	94.37
0.5	0.6	6.87	93.12
0.6	0.9	9.37	90.62
0.1	0.5	3.75	96.25
0.3	0.6	5.62	94.37
0.5	0.7	7.5	92.5
0.5	0.7	7.5	92.5
0.3	0.9	7.5	92.5
0.3	0.6	5.62	94.37
0.3	0.7	6.25	93.75
0.5	0.9	8.75	91.25
0.2	0.5	4.37	95.62
0.1	0.8	5.62	94.37
0.4	0.6	6.25	93.75
0.2	0.6	5	95
0.5	0.7	7.5	92.5
		6.191	93.81
		Error promedio %	Efectividad promedio %

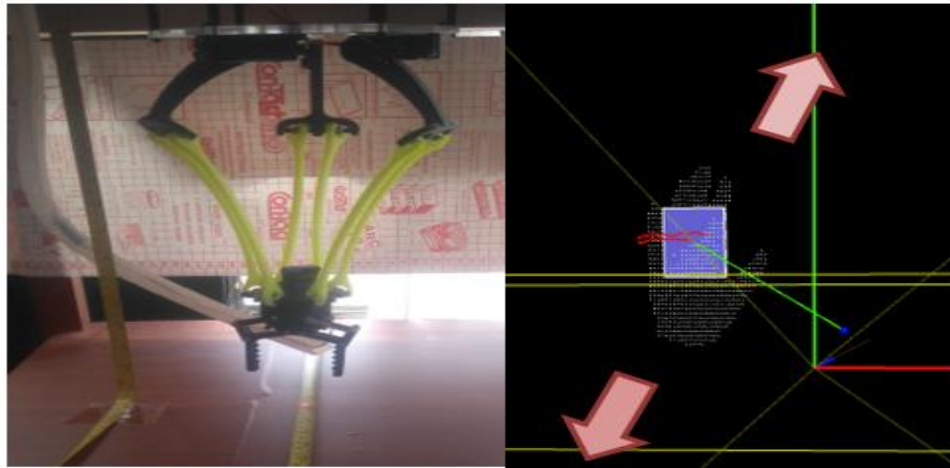
Tabla de valores de acierto y error en el movimiento Horizontal.

Elaborado por: Esteban Martínez

4.3.3 Movimiento de Profundidad

Segunda prueba básica que hace referencia al movimiento de la mano en el eje Z, se procedió a mover la mano de adentro hacia afuera, para posteriormente documentar las medidas obtenidas, Figura 4.9.

Figura 4.9 Protocolo de pruebas de movimiento en el eje Z



Pruebas de movimiento de profundidad. Elaborado por: Esteban Martínez

Tabla 4.5 Valores obtenidos en la trayectoria trazada en el eje Z.

Trayectoria vertical dada por el patrón que sigue la mano vs Trayectoria vertical imitada por el robot Delta				
Movimiento en eje Z punto inicial de la trayectoria y punto final de la trayectoria				
Número de datos tomados	Movimiento de la mano		Movimiento del robot	
	Punto inicial	Punto final	Punto inicial	Punto final
1	13	41	13.8	39.9
2	13	41	13.7	40
3	13	41	13.7	39.8
4	13	41	13.9	40
5	13	41	13.8	40
6	13	41	13.9	39.9
7	13	41	13.9	40
8	13	41	13.7	39.9
9	13	41	14	39.8
10	13	41	13.7	39
11	13	41	14	39.8
12	13	41	13.8	40
13	13	41	14	40
14	13	41	13.7	39.9
15	13	41	14	39.8
16	13	41	13.7	39.7
17	13	41	13.8	39.9
18	13	41	13.8	40
19	13	41	14	39.8
20	13	41	14	39.7

Tabla de valores en el movimiento de profundidad. Elaborado por: Esteban Martínez

Tabla 4.6 Tabla de efectividad y errores en la trayectoria trazada en el eje Z.

Trayectoria vertical dada por el patrón que sigue la mano vs Trayectoria vertical imitada por el robot Delta			
Movimiento en eje Z punto inicial de la trayectoria y punto final de la trayectoria			
Tabla de resultados			
Error en Punto inicial	Error en Punto final	Error total %	Efectividad total %
0.8	0.9	6.54	93.46
0.7	1	6.54	93.46
0.7	0.8	5.77	94.23
0.9	1	7.31	92.69
0.8	1	6.92	93.08
0.9	0.9	6.92	93.08
0.9	1	7.31	92.69
0.7	0.9	6.15	93.845
1	0.8	6.92	93.08
0.7	0	2.62	97.31
1	0.8	6.92	93.08
0.8	1	6.92	93.08
1	1	7.69	92.31
0.7	0.9	6.15	93.87
1	0.8	6.92	93.08
0.7	0.7	5.38	94.62
0.8	0.9	6.54	93.46
0.8	1	6.92	93.08
1	0.8	6.92	93.08
1	0.7	10.63	93.47
		6.70	93.5
		Error promedio %	Efectividad promedio %

Tabla de valores de acierto y error en el movimiento de profundidad.

Elaborado por: Esteban Martínez

4.4 Pruebas de precisión de empaqueo de objetos.

Las siguientes 3 pruebas se las hará de tal manera que la mano del operario siga una trayectoria específica dada, es decir que cumpla un patrón de movimiento durante 20 intentos, el objetivo de la prueba es colocar una pelota de tenis de mesa en su empaque colocado en diferentes lugares del área de trabajo del robot Delta para determinar la precisión.

4.4.1 Prueba de precisión de empackado en un punto X1 del área del trabajo

Figura 4.10 Prueba de precisión 1 en un punto del área de trabajo



Pruebas de movimiento de acierto en un punto X1 del área de trabajo.

Elaborado por: Esteban Martínez

Tabla 4.7 Valores obtenidos en La prueba de precisión punto X1

Trayectoria mediante un patrón que sigue la mano vs Trayectoria imitada por el robot Delta		
Prueba de acierto en el empaque de pelotas de Tenis de mesa colocada en un punto x1 del área de trabajo		
Número de pruebas	Movimiento del robot	
	Prueba acertada	Prueba fallida
1	1	0
2	1	0
3	1	0
4	1	0
5	1	0
6	1	0
7	0	1
8	1	0
9	1	0
10	1	0
11	1	0
12	1	0
13	1	0
14	0	1
15	1	0
16	1	0
17	1	0
18	1	0
19	1	0
20	0	1

Tabla de resultados en el movimiento de precisión X1. Elaborado por: Esteban Martínez

Tabla 4.8 Tabla resumen de porcentajes de resultado de prueba 1

Trayectoria mediante un patrón que sigue la mano vs Trayectoria imitada por el robot Delta	
Prueba de acierto en el empaque de pelotas de Tenis de mesa colocada en un punto x1 del área de trabajo	
Movimiento del robot	
Total, de pruebas acertadas	Total, de pruebas fallidas
17	3
Efectividad total promedio %	Error total promedio %
85	15

Tabla de resultados resumen en el movimiento de precisión área X1.

Elaborado por: Esteban Martínez

Para el cálculo de efectividad en las pruebas de acierto en un punto del robot Delta se la ha realizado con la Ecuación 4.7, mientras que para el cálculo de error se la ha realizado con la Ecuación 4.8 descritas a continuación.

$$Efectividad\ total\ promedio = \frac{Total\ de\ pruebas\ acertadas}{Numero\ de\ pruebas\ totales} \times 100\% \quad Ec. (4.7)$$

$$Error\ total\ promedio = \frac{Total\ de\ pruebas\ fallidas}{Numero\ de\ pruebas\ totales} \times 100\% \quad Ec. (4.8)$$

4.4.2 Prueba de precisión de empackado en un punto X2 del área del trabajo

Figura 4.11 Prueba de precisión 2 en un punto del área de trabajo



Pruebas de movimiento de acierto en un punto X2 del área de trabajo.

Elaborado por: Esteban Martínez

Tabla 4.9 Valores obtenidos en La prueba de precisión punto X2

Trayectoria mediante un patrón que sigue la mano vs Trayectoria imitada por el robot Delta		
Prueba de acierto en el empaque de pelotas de Tenis de mesa colocada en un punto x2 del área de trabajo		
Número de pruebas	Movimiento del robot	
	Prueba acertada	Prueba fallida
1	0	1
2	1	0
3	0	1
4	1	0
5	1	0
6	1	0
7	1	0
8	1	0
9	1	0
10	1	0
11	1	0
12	1	0
13	1	0
14	0	1
15	0	1
16	1	0
17	1	0
18	1	0
19	1	0
20	1	0

Tabla de resultados en el movimiento de precisión X2.

Elaborado por: Esteban Martínez

Tabla 4.10 Tabla resumen de porcentajes de resultado de prueba 2

Trayectoria mediante un patrón que sigue la mano vs Trayectoria imitada por el robot Delta	
Prueba de acierto en el empaque de pelotas de Tenis de mesa colocada en un punto x2 del área de trabajo	
Movimiento del robot	
Total, de pruebas acertadas	Total, de pruebas fallidas
16	4
Efectividad total promedio %	Efectividad total promedio %
80	20

Tabla de resultados resumen en el movimiento de precisión área X2.

Elaborado por: Esteban Martínez

4.4.3 Prueba de precisión de empaçado en un punto X3 del área del trabajo

Figura 4.12 Prueba de precisión 3 en un punto del área de trabajo



Pruebas de movimiento de acierto en un punto X3 del área de trabajo.

Elaborado por: Esteban Martínez.

Tabla 4.11 Valores obtenidos en La prueba de precisión punto X3

Trayectoria mediante un patrón que sigue la mano vs Trayectoria imitada por el robot Delta		
Prueba de acierto en el empaque de pelotas de Tenis de mesa colocada en un punto x3 del área de trabajo		
Número de pruebas	Movimiento del robot	
	Prueba acertada	Prueba fallida
1	1	0
2	1	0
3	1	0
4	1	0
5	1	0
6	0	1
7	1	0
8	1	0
9	1	0
10	1	0
11	1	0
12	0	1
13	0	1
14	1	0
15	1	0
16	1	0
17	1	0
18	0	1
19	1	0
20	1	0

Tabla de resultados en el movimiento de precisión x3. Elaborado por: Esteban Martínez

Tabla 4.12 Tabla resumen de porcentajes de resultado de prueba 3

Trayectoria mediante un patrón que sigue la mano vs Trayectoria imitada por el robot Delta	
Prueba de acierto en el empaque de pelotas de Tenis de mesa colocada en un punto X3 del área de trabajo	
Movimiento del robot	
Total, de pruebas acertadas	Total, de pruebas fallidas
16	4
Efectividad total promedio %	Efectividad total promedio %
80	20

Tabla de resultados resumen en el movimiento de precisión área X3.

Elaborado por: Esteban Martínez

Tabla 4.13 Tabla resumen de porcentajes de resultado de las 3 pruebas de acierto en
un solo punto.

Movimiento del robot	
Efectividad total %	Error total %
81.67	18.33

Tabla de resultados resumen en el movimiento de precisión área X1, X2 y X3.

Elaborado por: Esteban Martínez

Para el cálculo de efectividad total en las pruebas de acierto en los 3 sitios diferentes en los que actuó el robot Delta se la ha realizado con la Ecuación 4.9, mientras que para el cálculo de error total se la ha realizado con la Ecuación 4.10 descritas a continuación.

$$Efectividad\ Total = \sum \left(\frac{Efectividad\ total\ promedio}{Numero\ de\ pruebas\ realizadas} \right) \times 100\% \quad Ec. (4.9)$$

$$\% Error\ Total = \sum \left(\frac{Error\ total\ promedio}{Numero\ de\ pruebas\ realizadas} \right) \times 100\% \quad Ec. (4.10)$$

CONCLUSIONES

Se determinó la cinemática inversa como el mejor método para la movilización y maniobrabilidad de la plataforma Delta, debido a que el objetivo era encontrar los valores que deben tener las articulaciones del robot una vez que se obtenga las coordenadas del brazo del usuario mediante el Sensor Kinect. y se confirmó la coherencia del movimiento de las miembros del robot con este procedimiento,

Se construyó el robot Delta bajo las especificaciones planteadas en los objetivos, el sistema realizado permite al usuario reflejar los movimientos de la mano capturados por el sensor hacia el robot en el espacio de trabajo, aunque es reducido debido a su tamaño limitado logró transportar objetos de un punto a otro, con lo cual se comprobó su funcionamiento tal como se muestra en la Tabla 4.13.

Se implementó un algoritmo de control en Processing mediante visión artificial utilizando la captura de movimientos de la mano de una persona obtenidos por una plataforma controladora de videojuegos tan versátil como lo es el sensor Kinect V1.14 para operar el movimiento del robot Delta utilizando las librerías presentes en el entorno de programación junto con las investigaciones actuales de control a distancia del entorno de Processing.

Se comprobó la precisión de la aplicación desarrollada en el robot Delta con los experimentos de posicionamiento de pelotas de tenis llevadas desde un punto del área de trabajo hacia su respectivo envase, alcanzando un índice de acierto del 81.67% tal como muestra la Tabla 4.13 en las pruebas de acierto en un solo punto

El error producido en las pruebas de acierto en un solo punto es del 18.33% y este corresponde al sensor Kinect V 1.14, debido al ingreso de medidas inconstantes en el algoritmo de control cuando se obtienen la postura de la mano ocasionando variación en su movimiento, en parte debido a la versión de hardware y librerías desactualizadas del sensor pudiendo ser solucionada usando una versión más actual del modelo de Kinect.

El Kinect V1.14 a pesar de ser un modelo antiguo de sensor de Microsoft permitió la labor de operación en conjunto con el diseño implementado para la construcción del

robot Delta ya que se hizo un enfoque correcto de recursos de las librerías de comunicación con el software de Processing aprovechando al máximo los elementos presentes en este modelo del sensor Kinect.

Se hizo un uso adecuado de las librerías tipo gesture y cloud points presentes en el entorno de Processing usados en visión artificial cumpliendo con la meta de manipular y posicionar objetos en un lugar determinado a pesar de que esta versión del sensor no posee identificación de la mano en los puntos de censado del esqueleto capturado por Kinect.

La mejor opción para la manipulación de objetos en un espacio de trabajo es un robot que muestre rigidez y precisión elevada al momento de colocarlos, los resultados mostrados en los experimentos de movimiento del robot Delta cumplieron con los requerimientos establecidos, lo cual lo hacen un candidato idóneo para el posicionamiento de objetos.

RECOMENDACIONES

Para una mejor recepción de captura de movimientos por parte del sensor Kinect V1.14 Xbox 360 es recomendable tener la cámara en un espacio abierto y ubicada en un entorno en el cual no actúen cuerpos, o elementos externos ajenos cuando se procede a capturar los movimientos de la mano.

El sensor para Xbox 360 Kinect V1.14 es uno de los primeros sensores que salió al mercado, por lo cual el modelo se encuentra bastante desactualizado, para futuras optimizaciones de algoritmos de control y una mayor precisión de datos capturados se recomienda utilizar versiones más recientes del sensor, las versiones más actuales del sensor permiten utilizar una mayor cantidad de librerías actualizadas que están orientadas al desarrollo de aplicaciones de control aportando una mayor versatilidad al momento de elegir un algoritmo de manera que exista un control adecuado y eficiente en las aplicaciones de manipulación de objetos.

Se recomienda trabajar con la versión 2.2.1 de Processing, a pesar de ser una versión antigua se denota el deficiente funcionamiento de la comunicación entre el sensor Kinect y el entorno de Processing si se usan versiones anteriores a esta.

Para la instalación y reconocimiento entre sensor Kinect V1.14 y el gestor de paquetes de comunicación presente en la computadora Apple es necesario seguir un orden y se recomienda seguirlo ya que si no se continúa un orden cronológico se presentan inconsistencias entre la ejecución del programa, el orden de instalación es el siguiente:

- Openni-MacOsx-1.5.4.0
- Sensor-MacOsx-5.1.2.1
- Nite-MacOsx-1.5.4.0

Para la construcción del robot es recomendable asesorarse con una persona experta en el tema de fabricación por método de impresión 3D, ya que así se puede avanzar simultáneamente en la selección de elementos electrónicos que conlleva a la circuitería y en el diseño de la estructura mecánica de tal forma que no existan inconvenientes por falta espacio y confusiones en el cableado.

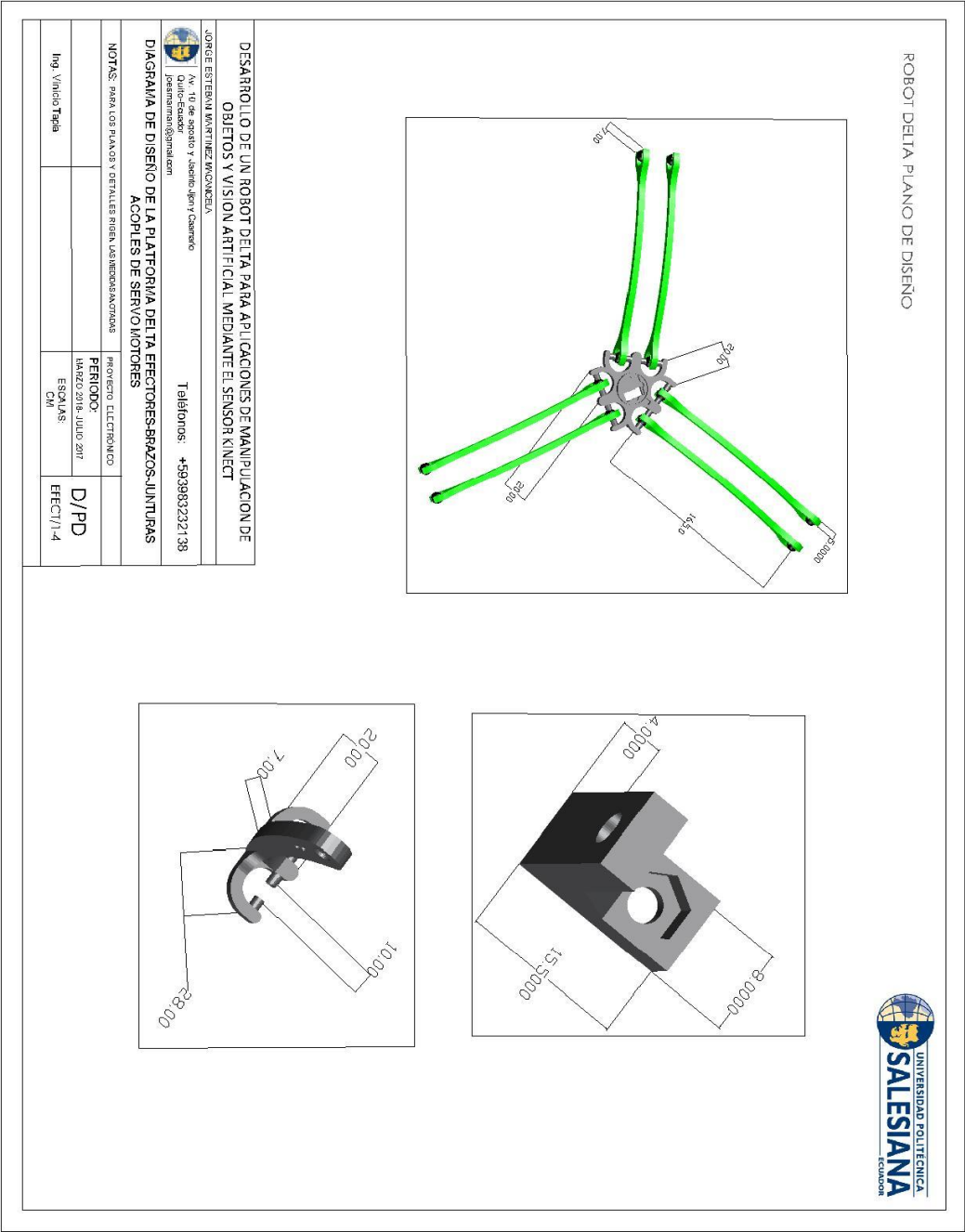
REFERENCIAS BIBLIOGRÁFICAS

- Ben Fry, C. R. (2015). *Getting Started with Processing, 2nd Edition*. Los Angeles : Maker Media, Inc.
- Doumanoglou, A. (2013). A dataset of Kinect-based 3D scans. *IEEE*, 1-30.
- Flacco Fabrizio, A. D. (2017). Real-Time Computation of Distance to Dynamic Obstacles With Multiple Depth Sensors. *IEEE Robotics and Automation Letters*, 56-63.
- Goudsmith. (2013). Drive magnetism. En Goudsmith, *Magnet grippers* (pág. 4). Petunialaan: Goudsmith Magnetic Systems.
- IEEE. (2015). Robust control for a delta robot. *International Conference on Advanced Robotics (ICAR)* (págs. 1-25). Akita, Japan: IEEE.
- Isidro Zabalza, J. R. (25 de Octubre de 2007). APLICACIONES ACTUALES DE LOS ROBOTS PARALELOS . Pamplona, Navarra, España.
- López-Betancourt-Martínez-Ancira-Ramirez. (23 de Septiembre de 2011). DISEÑO Y CONSTRUCCIÓN DE UN ROBOT HEXÁPODO. *MEMORIAS DEL XVII CONGRESO INTERNACIONAL ANUAL DE LA SOMIM*, págs. 310-317.
- Ojosnegros Ramos, E. (1 de Septiembbre de 2014). DESARROLLO DE UN SISTEMA PARA LA MONITORIZACIÓN DE OPERACIONES MANUALES MEDIANTE SENSOR KINECT . Valladolid, Valladolid, España.
- Pillajo, C., & Sierra, J. (2013). Human Machine Interface HMI using Kinect sensor to control a SCARA Robot. *2013 IEEE Colombian Conference on Communications and Computing (COLCOM)* (págs. 1-5). Medellin, Colombia: IEEE.
- Pose, A. C. (1 de Junio de 2015). ANÁLISIS CINEMÁTICO Y DINÁMICO DE PINZA DE ROBOT PARA ESPACIOS AISLADOS. Leganes, Madrid, España.

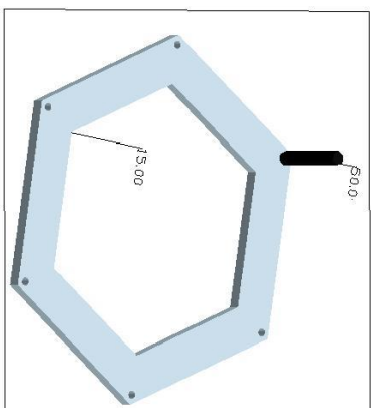
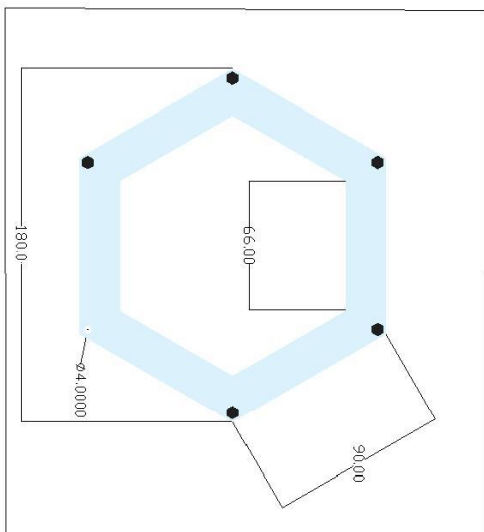
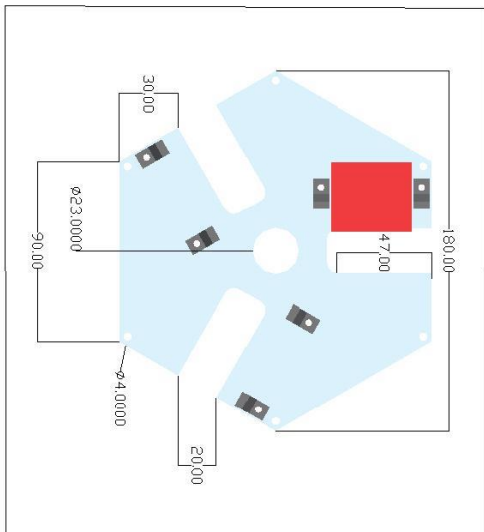
- Ralph-Christoph Weber, E. O. (2011). *Festo Didactic*. Denckendorf: Festo Didactic GmbH & Co.
- Rick Waldron. (2015). *Make: JavaScript Robotics*. San Francisco: Maker Media, Inc.
- Soriano, E. (2015). Diseño y análisis de una pinza de agarre con dedos retráctiles. *Research Gate*, 1-8.
- V. Poppeová, J. U. (s.f.). DELTA ROBOTS. *ROBOTS FOR HIGH SPEED MANIPULATION*. University of Žilina, Žilina, Slovakia.
- Vishwakarma, S. K. (2017). Caracterización del material del ABS. *quest Revistas*, 13-16.
- Vivas, A. (7 de 05 de 2013). *ROBÓTICA PARALELA: APLICACIONES INDUSTRIALES, MODELADO Y CONTROL*. Cauca, Popayán, Colombia.
- Williams, R. L. (8 de Enero de 2016). *The Delta Parallel Robot: Kinematics Solutions*. Columbus, Ohio, EEUU.
- Zsombor-Murray, P. (1 de Abril de 2004). Descriptive Geometric Kinematic Analysis of Clavel's "Delta" Robot. 6. Quebec, Montreal, Canada: Department of Mechanical Engineering.

ANEXOS

Anexo A Dimensiones de diseño



ROBOT DELTA PLANO DE DISEÑO



DESARROLLO DE UN ROBOT DELTA PARA APLICACIONES DE MANIPULACION DE OBJETOS Y VISION ARTIFICIAL MEDIANTE EL SENSOR KINECT

JORGE ESTEBAN MARTINEZ WACANDELA
Av. 10 de agosto y Jazeto, libre y Camarillo
Quito Ecuador
jmartinez@upol.edu.ec

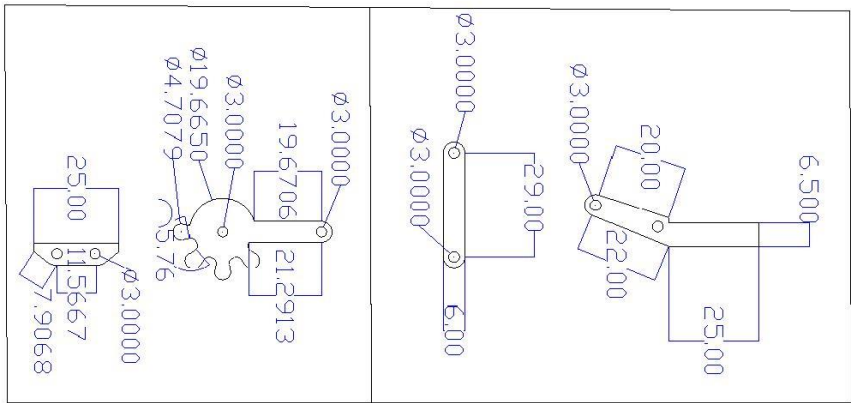
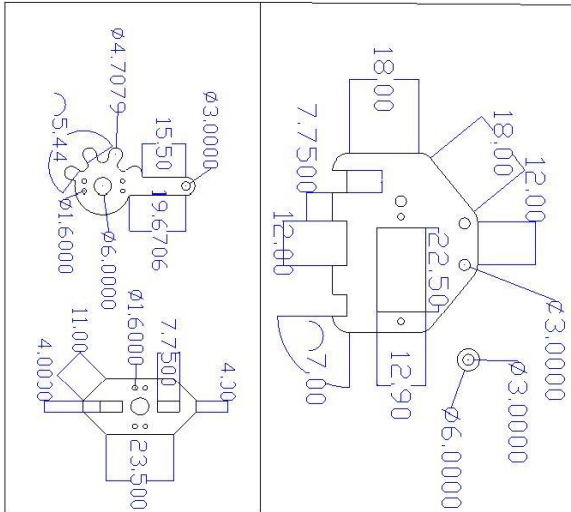
Teléfonos: +593983232138

DIAGRAMA DE DISEÑO DE LA PLATAFORMA DELTA BASE MOVIL

NOTAS: PARA LOS PLANOS Y DETALLES RELEN LAS MEDIDAS SIGUIENTES

Ing. Vinicio Tapa	PERIODO: Escala mm	PROYECTO ELECTRONICO MARZO 2016- JULIO 2017	D/PD BASE/24
-------------------	--------------------------	--	-----------------

ROBOT DELTA PLANO DE DISEÑO



DESARROLLO DE UN ROBOT DELTA PARA APLICACIONES DE MANIPULACION DE OBJETOS Y VISION ARTIFICIAL MEDIANTE EL SENSOR KINECT

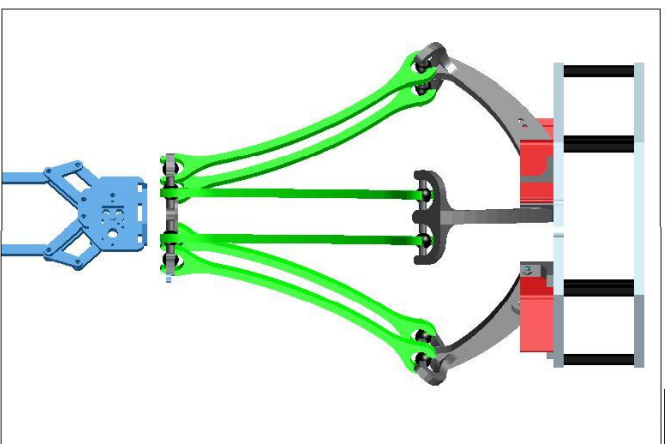
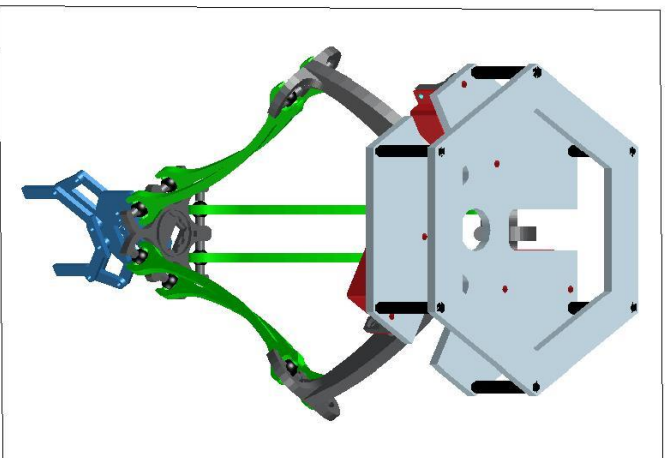
JORGE ESTEBAN MARTINEZ INCAKUNELA
Av. 10 de agosto y Jicšto, Jicšto y Camino
Quito Ecuador
jmartinez@upsa.edu.ec
Teléfonos: +593983222138

DIAGRAMA DE DISEÑO DE LA PLATAFORMA DELTA PINZA

NOTAS: PARA LOS PLANOS Y DETALLES RELEN LAS MEDIDAS ANTERIORES

PERIODO: ELECTRONICO

Ing. Vinicio Tapa
Escala: 1:1
Base: 3-4



DESARROLLO DE UN ROBOT DELTA PARA APLICACIONES DE MANIPULACION DE OBJETOS Y VISION ARTIFICIAL MEDIANTE EL SENSOR KINECT

JORGE ESTEBAN MARTINEZ MACANCELA
Av. 10 de agosto y Jazmín Jilfray Camino
Quito-Ecuador
jmartinez@upoli.edu.ec
Teléfonos: +593983232138

DIAGRAMA DE DISEÑO DE LA PLATAFORMA DELTA PINZA

NOTAS: PARA LOS PLANOS Y DETALLES REVER LAS VENTAS ANTERIORES

Ing. Vinicio Tapa	PERIODO: FEBRERO 2018- JULIO 2017	D/PD
	ESCALAS mm	DELTA/44

Anexo B Librerías de Comunicación presentes en el entorno de Processing

```
* -----  
-----  
  
import processing.opengl.*; //Pseudocodigo utilizado para variables  
import kinectOrbit.*; //Pseudocodigo utilizado para librerias  
KinectOrbit kOrbit;  
void setup() {  
    size(800, 600, OPENGLE);  
    smooth();  
    kOrbit = new KinectOrbit(this, 0, "kinect"); //verification VAR  
    kOrbit.drawGizmo(true); //  
    kOrbit.shiftControl(true); // //verification de ventanas  
}  
void draw() {  
    background(0);  
    kOrbit.pushOrbit(this);  
    box(50);  
    kOrbit.popOrbit(this); // End Orbiting  
  
* -----  
-----  
* -----  
-----
```

Para tener una mayor comprensión acerca de las librerías utilizadas en el entorno de programación processing se utilize la interfaz de OpenNI desarrollado por Max Reiner, ubicadas en la siguiente dirección web:

<https://structure.io/openni>

Anexo C Descripción de los materiales utilizados en el proyecto técnico.

Sensor Kinect V1.14

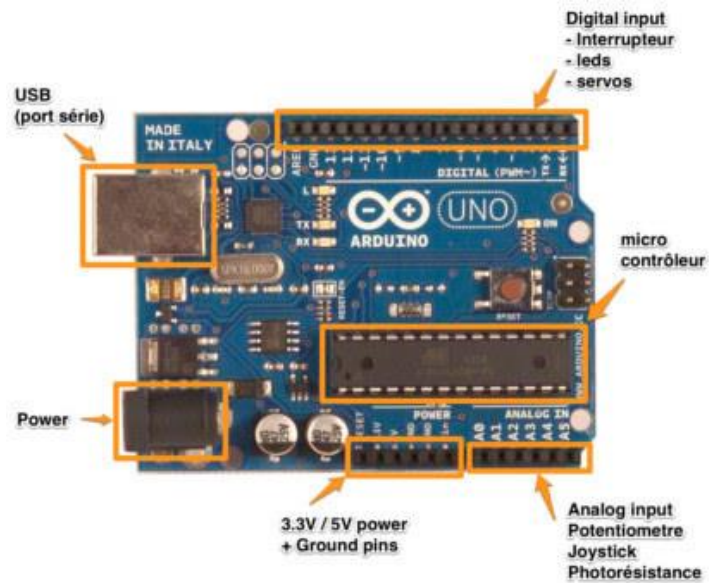


El sensor Kinect es un dispositivo creado por la empresa Windows para la consola de videojuegos Xbox, su funcionalidad la hizo una herramienta muy útil en el ámbito de los videojuegos que luego se popularizó más tarde como un controlador de interfaz abierta a programadores.

Para mayor información sobre el Sensor Kinect se puede referir a la siguiente página web:

https://upcommons.upc.edu/bitstream/handle/2117/23997/MICROSOFT_KINECT.pdf;jsessionid=7B12386E0A71C53E3E46D58C3820F55B?sequence=1

Tarjeta Arduino Uno



La tarjeta Arduino es una placa con orientaciones educativas, en los últimos la tecnología Arduino ha sido tan popular que ha incursionado en la industria, existe un modelo de Plc que se encuentra en desarrollo que utiliza como cerebro a la placa Arduino.

Para mayor información de la tarjeta puede referirse a esta página web:

<http://arduino.cl/arduino-uno/>

Plástico ABS (Acrilonitrilo Butadieno Estireno)

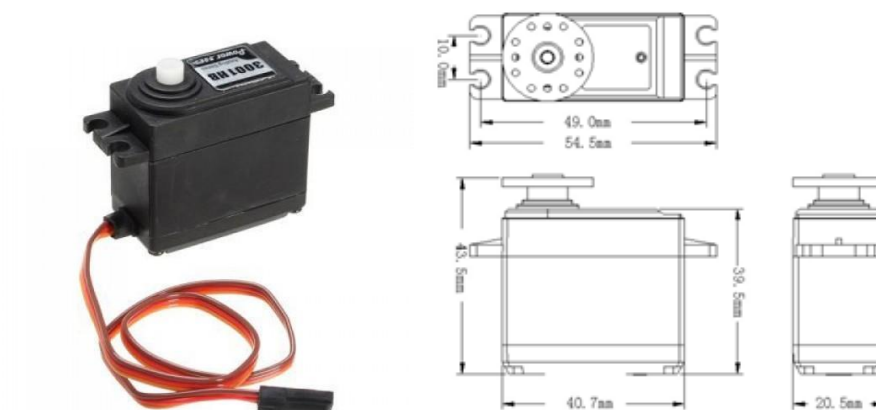


El plástico ABS más conocido como acrilonitrilo-butadieno-estireno, es un plástico muy duro y moldeable utilizado en el método de impresión 3D, debido a su bajo costo, y resistente al alto impacto

Para las especificaciones químicas acerca del Plástico ABS puede referirse a esta página:

https://repositorio.itesm.mx/bitstream/handle/11285/569376/DocsTec_1207.pdf?sequence=1

Servo Análogo HD-3002HB



Un servomotor muy utilizado en el mercado actual en el desarrollo de prototipos de robot orientados a la educación el número del servo motor es el siguiente: Servo Analog HD-3002HB.

Para las Especificaciones técnicas acerca de este servomotor puede referirse a la siguiente página web:

<https://servodatabase.com/servo/power-hd/hd-3001hb>